

öS 60,-/sfr 7,-/Lit 7000/hfl 8,50 DM7,–
AMIGA
Markt&Technik
4–'88
DAS COMPUTERMAGAZIN FÜR AMIGA - FANS
Grundlagen und Tests
Die ideale Datenbank
Exklusiv: Kurs und Compiler
Modula-2 die Sprache mit Zukunft
Abenteuerspiele im Test
Spiele mit Strategie
Titelgenerator und Genlock
Amiga für Video-Fans
Tolle Grafik-Listings zum Abtippen
★ Faszinierende Fraktalberge
★ C 64-Grafiken auf dem Amiga
A-L
M-Z
A-Z



1 Golem Drive 3,5
NEC 1036a mit heller Frontblende ● Amiga-farbenes Metallgehäuse ● Abschalter ● Busdurchführung bis DF3 ● PC-Karten und Sidecar kompatibel **!!! neu !!! jedes Drive mit Trackdisplay** zur aktuellen Spur- und Kopfanzeige

mit Display — DM **379.-**
ohne Display — DM **369.-**

2 Golem Drive 5.25
5,25 Zoll Laufwerk mit heller Frontblende ● Amiga-farbenes Metallgehäuse ● Abschalter ● Busdurchführung bis DF3 ● 40/80 Track Umschalter Amiga und MS-Dos kompatibel **!!! neu !!! Drive mit Trackdisplay wie Golem 3,5**

mit Display — DM **449.-**
ohne Display — DM **439.-**

3 Golem Drive 3,5 intern
modifiziertes NEC 1036a mit heller Blende ● Staubschutzklappe zum Einbau in den A 2000 incl. Einbausatz und Einbauanleitung — DM **269.-**

4 Golem Ram Box 1000
2 MB Erweiterung ansteckbar ● in Amiga-farbenem Metallgehäuse ● Abschalter ● Busdurchführung ● auto konfigurierend ● Betriebskontrollanzeige durch LED ● erweitert den Grundspeicher auf 2,5 Megabyte

DM **998.-**

Technische Änderungen vorbehalten

5 Golem 500
Ram Erweiterung speziell für den Amiga 500 ● technische Einzelheiten wie Golem Ram Box 1000 ● beide Erweiterungen ohne Wait States

DM **998.-**

6 Kickstart / Uhr Modul
"Bitte Workbench einlegen", so meldet sich ihr Amiga 1000 mit dem Kickstart Eprom Modul ● Ansteckbar am Systembus ● Amiga-farbenes Metallgehäuse ● durchgeführter Systembus ● abschaltbar sodaß andere Kickstart Versionen wieder gebootet werden können.

DM **199.-**

Amiga 2000 u. 500 kompatibeles, externes Uhrenmodul ansteckbar am Systembus ● Software, die die 2000/500 Uhr anspricht, benutzt auch die Golem Clock für den A 1000

DM **149.-**
Uhr und Kickstart in einem Gehäuse — DM **299.-**

7 Golem Sound
Audio Digitizer der Spitzenklasse, kompatibel zu aller gängigen Software mit DIN- und Cinch-Anschluß auch für Micro Anschluß geeignet ● optische Aussteuerung über ein LED Display ● STEREO ● Wandlungsfähig ● 1MHz getaktet bietet der Golem Sound unglaubliche Sample Qualität.

Mono — DM **139.-**
Stereo — DM **189.-**
Software zum Golem Sound, stereofähig — DM **29.-**

Entwurf Firma DataCom Dortmund · Realisierung Plum 5014 Kerpen-Sindorf, Tel. 0 22 73/5 30 91

AMIGA wird ständig besser

Was haben Fußballprofis mit Computern im Sinn? Bisher lautete die Antwort eigentlich immer: eher wenig bis gar nichts. Um so erstaunlicher erscheint es jetzt, daß sich die hochbezahlten Kicker des F.C. Bayern München einer abendlichen Fortbildungsmaßnahme in Sachen Computer unterziehen. Hatten Sie bisher wenig Beziehung zum Inhalt der Werbung auf ihrer Brust (Commodore), versuchen Sie derzeit, mit Hilfe der AMIGA-Redaktion, den Umgang mit einem neuen Medium zu erlernen. Einige kluge Köpfe beim F.C. Bayern und bei Commodore hatten erkannt, daß es für die Profis auch ein »Leben nach dem Fußball« gibt. Und in der dann sich öffnenden Berufswelt bestimmen die Computer immer mehr das Bild. Wer heute nicht den Anschluß behält, steht eventuell morgen schon im Abseits.

Abseitig ist daher sicherlich nicht der Wunsch der Spieler, aber auch deren Frauen, sich bei Zeiten dem Computer und im speziellen dem Amiga zuzuwenden. Dies könnte möglicherweise mehrere positive Effekte auf die gesamte Entwicklung der Einstellung zu den Computern in weiten Teilen der Bevölkerung haben. Denn für viele Jugendliche, die bisher den Computer als Spiel oder Teufelszeug der Intelligenzbestien ansahen, treten die Idole des Sports als Vorreiter einer neuen Einsicht auf.

Der Erkenntnis nämlich, daß viele Dinge zwar nach wie vor ohne Computer gehen, es aber wesentlich bequemer und intelligenter ist, sich der elektronischen Rechenhilfen zu bedienen. Daß es auch noch Spaß machen kann, sich mit dem Computer zu beschäftigen, zeigt der Bericht über das »Bayern Training« ab Seite 8.



Der Torwart vom F.C. Bayern ist vom Amiga begeistert

Für die Redaktion des AMIGA-Magazins ist es eine besondere Freude, sich für diese Schulung einzusetzen. Stellt es doch eine Art »Fronteinsatz« für die spezialisierten Wissensvermittler, sprich Redakteure, dar. In diesen abendlichen Seminaren erlangen wir unter anderem die Kenntnis, wo den Einsteiger auf dem Amiga der Schuh drückt, worauf wir also auch im AMIGA-Magazin verstärkt unser Augenmerk richten müssen. Und das kommt Ihnen letztendlich zugute — in noch besser durchdachten und auf Ihre Belange abgestimmten Beiträgen.

Eine weitere Verbesserung werden Sie vielleicht in dieser Ausgabe bereits feststellen. Wenn Sie genau hinsehen, so hat sich die eine oder andere Änderung in der optischen Präsentation der einzelnen Beiträge ergeben. Auch auf diesem Gebiet haben wir dazugelernt, beziehungsweise zum erstenmal einem »Art Director« die gestalterische Verantwortung für das AMIGA-Magazin übergeben. Das heißt wir verbessern unsere Testberichte, Kurse, Listings und Aktuellmeldungen nicht nur permanent vom Inhalt her, sondern oder gerade auch die optische Aufmachung. Das Erscheinungsbild der AMIGA wird sich in der nächsten Zeit konsequent in Richtung »für Auge und Hirn noch besser« entwickeln. Dafür arbeiten wir derzeit intensiv an allen Ecken und Enden, und das im wahrsten Sinne des Wortes, an Fortbildungsmaßnahmen für Ihre AMIGA-Zeitschrift. Wenn Sie die AMIGA, die Sie gerade in den Händen halten, mit den vorherigen Ausgaben vergleichen, werden Sie die Unterschiede bereits bemerken.

Gemausert hat sich auch unsere aktuelle Berichterstattung zu den neuen Produkten für den Amiga. Nicht nur Gerüchte, sondern handfeste Hintergrundinformationen, beleben die Rubrik »Aktuelles«.

Sie sehen, es tut sich einiges in Ihrem AMIGA-Magazin. Doch eines kann ich Ihnen versprechen, trotz oder gerade mit Änderungen werden wir immer die Inhalte in der Verpackung, die Sie schätzen, liefern. In der einen Ausgabe mal mehr, in der anderen mal weniger. Es ist so, zumindest glauben wir das, für jeden immer etwas dabei. Ganz gleich, ob Sie Einsteiger oder Profi sind, ob Sie an Neuigkeiten oder Hintergrundwissen interessiert sind, wir liefern Ihnen die Informationen, die Sie zum Amiga brauchen.

Über Anregungen und konstruktive Kritik, die zur Verbesserung der AMIGA beitragen, freuen wir uns immer.

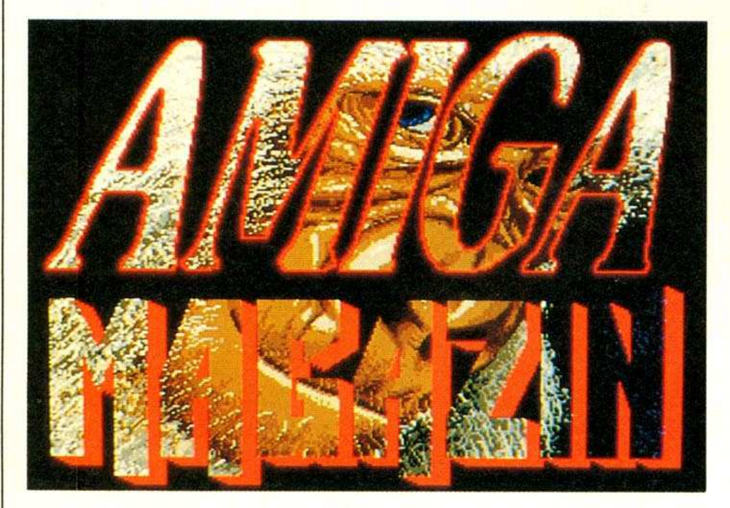
Herzlich Ihr Chefredakteur
Albert Absmeier



Professionelle Video-Titel

Wenn Sie gut aussehende Titel und Vorspänne für Ihre Filme auf heimischen Videokassetten suchen, könnte der Video-Titler von Aegis das richtige für Sie sein. Lesen Sie den ausführlichen Test

ab Seite 136



AKTUELL

CeBIT '88 11

News im Überblick 13

EINSTEIGER

Grundlagen und Tests
Die ideale Datenbank
Das 1 x 1 der Dateiverwaltung AMIGA test 18

DAS AMIGA-GLOSSAR, Teil 8 (ST–ZO) 79

SPEICHER MIT ZUKUNFT 144

DATEIVERWALTUNG

DIE IDEALE DATENBANK 24

KURSE

C-Kurs für Anfänger (Teil 6) 90

Exklusiv: Kurs und Compiler
Modula-2 die Sprache mit Zukunft
Modula-2: Der Kurs stimmt (Teil 1) 98

TIPS & TRICKS

Nüsse und Austern 104

Patchen mit System 108

PUBLIC DOMAIN

Die Public Domain-Seite 112

SOFTWARE-TEST

Nimm M2... AMIGA test 114

Titelgenerator und Genlock
Amiga für Video-Fans
TITEL MIT PFIFF AMIGA test 136

Workbench-Renovierung AMIGA test 140

Aus den Tasten gekitzelt AMIGA test 142

Die etwas andere Datenbank AMIGA test 146

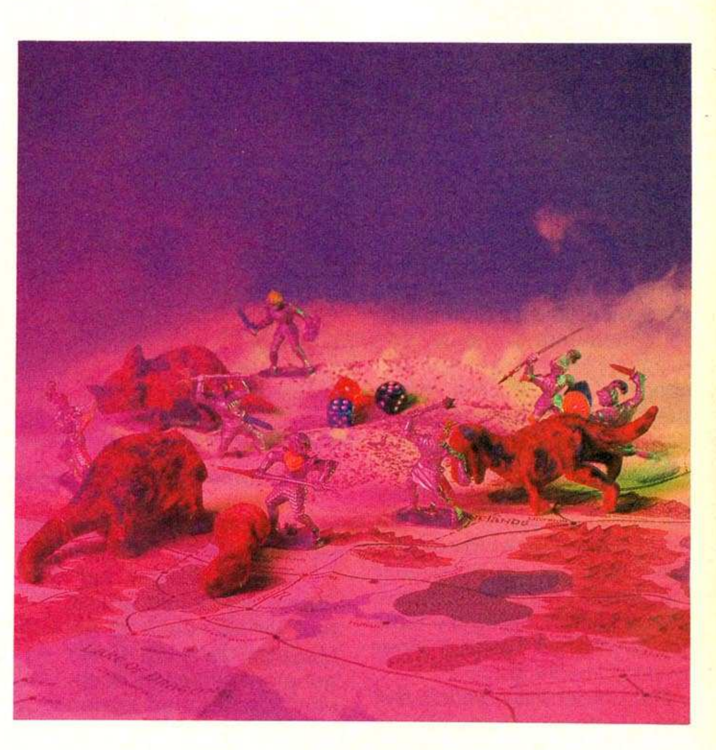
HARDWARE

Diskettenstation im Eigenbau 118

Der Weg in die Freiheit 122

ROLLENSPIELE

Die Welt der Phantasie 126



Wie geht man an die Rolle

Für alle, die sich für Rollenspiele interessieren, aber bisher wegen der Komplexität noch davor zurückgeschreckt sind, geben wir in einem umfassenden Artikel Hilfe.

Ab Seite 126

INHALT 4/88



Die ideale Datenbank

Dateiverwaltung ist ein hochinteressantes Thema. Wir vergleichen Superbase, Datamat, GoAmiga und Organize mit unserer fiktiven Datenbank DB2000.

Ab Seite 24

SPIELE-TEST

Abenteuerspiele im Test
Spiele mit Strategie
Das Schloß der Schatten…
…oder die Sache mit dem Glück — AMIGA test 130
Die Schlacht um Scandor — AMIGA test 134

HARDWARE-TEST

Kombiniere, Desktop Video — AMIGA test 148
Der Verwandlungskünstler — AMIGA test 150
Schnellzug — AMIGA test 151

AUFRUFE UND WETTBEWERBE

Gesucht: Tips & Tricks zum Amiga 149
Ihr Programm im AMIGA-Magazin 151

RUBRIKEN

Editorial 5
Impressum 16
Leserforum 36
Bücher 111
Programmservice 153
Vorschau 155
Inserentenverzeichnis 156

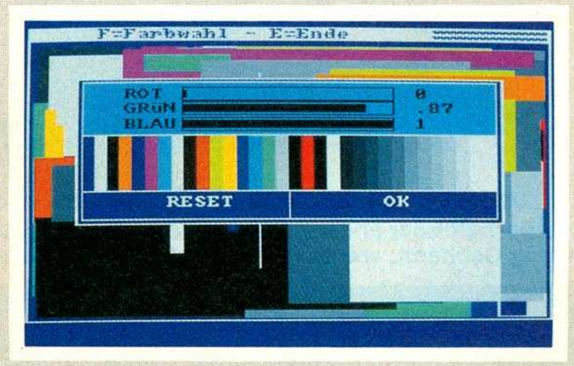
Dieses Symbol zeigt an, welche Programme auf Diskette erhältlich sind.

LISTINGS ZUM ABTIPPEN

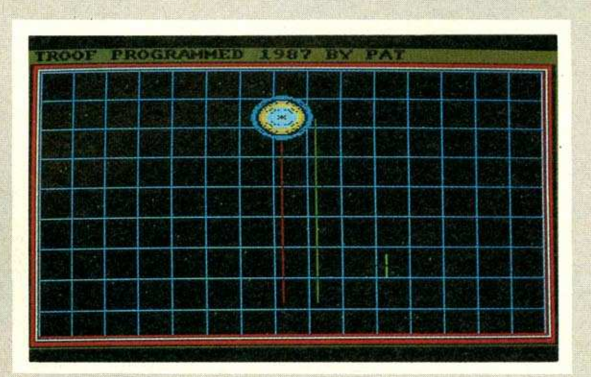
Tolle Grafik-Listings zum Abtippen
*** Faszinierende Fraktalberge**
Die Alpen auf dem Bildschirm 40
Schneller Überblick 48
Der Pseudo-Guru 50
*** C 64-Grafiken auf dem Amiga**
Bilder-Recycling 52
Rechtschreibung einfach leichtgemacht! 63
Gefährliche Sportspiele 66
Spionage auf der Diskette 70
Funktionsliste 77
Wahr oder falsch? Das ist hier die Frage! 78
Farben ändern wie ein Chamäleon 83



Die Alpen auf dem Bildschirm in Minutenschnelle mit »Fraktalberge«



Farbeinstellung in Basic ist kein Problem mit »ColorChange«



Ein tolles Spiel mit starker Grafik in Amiga-Basic ist »Troof«

AMIGA trainiert



Bild 1. Andreas Brehme studiert die AMIGA, Jürgen Wegmann horcht währenddessen interessiert zu



Bild 2. Viel Spaß am Amiga und mit der Maus haben Jean-Marie und Carmen Pfaff



Bild 3. Norbert Eder erklärt hilfsbereit dem Masseur Wolfgang Gebhardt, wie es funktioniert



Bild 4. Gespannt lauschen Trainer Jupp Heynckes und Jürgen Wegmann den Ausführungen



Bild 5. Ist ja toll, was man alles machen kann, meint Torhüter Raimond Aumann zu Angela Aumann



Bild 6. Andreas und Pilar Brehme beobachten mit Erwartung das Geschehen auf dem Bildschirm

den F.C.BAYERN



Bild 7. Jupp Heynckes holt sich nur in Sachen Computer Rat beim Redakteur Ulrich Brieden



Bild 8. Renate Großhardt, Roland Wohlfahrt und Norbert Nachtweih folgen den Erklärungen von René Beaupoil



Bild 9. Hans und Silke Flick freuen sich über eine gelungene Lösung der schwierigen Aufgabe



Bild 10. Gisela und Dieter Hoeneß lernen zusammen mit Monika und Klaus Augenthaler

Die Spieler des Deutschen Meisters im Fußball machen sich fit für den Computer. ■ In Abendkursen lernen die Kicker des F.C. Bayern München den Commodore Amiga kennen. ■ Die Redaktion des AMIGA-Magazins trainiert alle zwei Wochen die Fußballprofis auf ihrem neuen und faszinierenden Medium Amiga 2000. ■ Welche Erfahrung machen die Spitzensportler in ihrer neuen Disziplin? ■ Lesen Sie selbst. ■ Wir berichten Ihnen monatlich in der AMIGA ausführlich über die Abende mit den Bayern.

Markt&Technik



Bild 11. Gerade Dipl.-Ing. Hans Pflügler kann den Amiga gut privat wie beruflich gebrauchen

AMIGA trainiert F.C. Bayern

Um für die Zeit nach ihrer aktiven Karriere als Fußballprofis gerüstet zu sein, absolvieren die Spieler des F. C. Bayern München im Parkhotel Ramada derzeit ein abendliches Fortbildungsprogramm am Amiga. Die Aufgabe, den Ballkünstlern die nötige Fingerfertigkeit an der Tastatur des Amigas zu vermitteln, hat die Redaktion der Zeitschrift AMIGA übernommen.

Dieter Hoeneß, zuständig für die Sportpromotion bei Commodore, ist einer der Initiatoren der Veranstaltung. Er selbst hat seine aktive Laufbahn im Sommer 1987 beendet. Von daher weiß er aus Erfahrung, wie wichtig es ist, sich auf das »Leben nach dem Fußball« vorzubereiten. Ein bedeutendes Element in der heutigen Berufswelt stellen die Computer dar. Über kurz oder lang kommt keiner mehr bei seinem (zukünftigen) Arbeitsplatz an diesen nützlichen Helfern vorbei. Das haben auch die Kicker des F.C. Bayern München rechtzeitig erkannt.

Ein Training der dritten Art

Unter diesem Aspekt war die Bereitschaft und Begeisterung bei den Spielern des deutschen Rekordmeisters groß. Sie kamen nicht allein zur Schulung, sondern brachten ihre Ehefrauen und Lebensgefährtinnen mit. Gerade die Damen waren vom Amiga angenehm überrascht.

Selbstverständlich war die gesamte Prominenz des F.C. Bayern München anwesend; neben Präsident Prof. Dr. Scherer auch Manager Uli Hoeneß. Dr. Scherer ist Dozent für Elektronische Datenverarbeitung, dennoch nahm er aktiv an der Schulung teil. Alles was beim Bayern-Sponsor Commodore Rang und Namen hat, gab sich ein Stelldichein: Vizepräsident Harald Speyer, Europa-Manager Winfried Hoffmann, Geschäftsführer Deutschland Heinz Wiening und Pressesprecher Gerold Hahn. Die Tagespresse und Nachrichtenagenturen schickten ihre Reporter und Fotografen, um von diesem ungewöhnlichen Training der Bayern zu berichten.

Nach theoretischen Grundlagen über die Computer an und für sich und den Amiga im speziellen, ging es dann handgreiflich zur Sache. Die verschiedenen Elemente der Workbench wurden bewußt mit der Maus erobert. Angefangen von der Zeiteinstellung über das Schreiben kleiner Notizen bis zum Verändern des Mauszeigers bereiteten die ersten Schritte auf dem Amiga nicht nur keine Probleme sondern auch noch großen Spaß, wie die Bilder auf der vorhergehenden Seite beweisen. Die Demonstration, daß die chaotische Adreßverwaltung per Hand und Büchlein durch den Amiga wesentlich vereinfacht und übersichtlicher gemacht werden kann, überzeugte.

Zum Abschluß vergnügten sich die Sportler noch mit dem von Ariolasoft gestifteten Spiel Championship Golf. Die tolle Grafikauflösung und die vielen Auswahlmöglichkeiten beeindruckten die Spieler, von denen einige Golf spielen.

Spaß mit Amiga

Die spontane Reaktion vieler Spieler, das sei ja alles kinderleicht, man werde das nächste Mal sicher wieder kommen, zeigt, daß mit dem Amiga ein bequem zu bedienender Computer auf dem Markt ist. Wir werden deshalb ab der nächsten Ausgabe mit einem Kurs für Einsteiger im AMIGA-Magazin beginnen. Die Erfahrungen und Anregungen, die die Redakteure bei der Schulung der Bayernspieler gewinnen, werden diesem Kurs zugute kommen. Steigen Sie gemeinsam mit den Bayernspielern in die faszinierende Welt des Amigas ein. (aa)



Bild 12. Ursula Jacob, Vizepräsident Harald Speyer und Dieter Hoeneß in lockerer Runde nach dem Vortrag



Bild 14. Unser Chefredakteur Albert Absmeier bei seinem Vortrag am Amiga



Bild 13. Prof. Dr. Fritz Scherer, Uli Hoeneß (vorne), im Hintergrund Schatzmeister Kurt Hegerich, Winfried Hoffmann, Vizepräsident Hans Schiefele und Gerold Hahn

Insgesamt sind acht Abendveranstaltungen geplant. Unterwiesen werden die Bayern-Spieler unter anderem in der Adreßverwaltung mit Superbase, in die Textverarbeitung mit Becker Text, im Zeichnen und Konstruieren mit Deluxe Paint II, in der Druckerhandhabung, in der Finanzplanung und im Musizieren. Abgeschlossen werden die Abende mit Spielen von Ariolasoft.

CeBIT '88

HANNOVER MESSE CeBIT '88 NEWS

Die CeBIT wird einige Trends aufzeigen, die auch für Amiga-Benutzer interessant sind. Informationen rund um den Amiga finden Sie auch auf dem Markt & Technik-Stand in Halle 7, D34/E41. Was es sonst noch Neues gibt, lesen Sie auf den nächsten Seiten.

Auf der CeBIT wird Star einige neue Drucker präsentieren. Da ist zum einen der Star LC-10 Colour, die farbige Version des LC-10. Einen Test dieses 9-Nadel-Druckers finden Sie bereits in der AMIGA 3/88. Im Bereich der 24-Nadel-Drucker ist Star ebenfalls aktiv. Der LC24-10 soll im Low-Cost-Bereich Marktanteile erobern. Der Drucker hat nach Angaben des Herstellers unter anderem folgende Eigenschaften:
— vier eingebaute LQ-Fonts. Weitere Schönschriften sind über IC-Karten nachrüstbar;
— zwei Emulationen (ESC/P, IBM Proprinter X24);
— Papier-Parkfunktion;
— Schubtraktor;
— Auflösung 360 x 360 dpi.

Der LC24-10 soll in der Schriftart »Draft Elite« eine Geschwindigkeit von 170 und in »Draft Pica« von 142 Zeichen pro Sekunde erreichen. Für LQ-Schrift gibt Star Geschwindigkeiten von 56 beziehungsweise 47 Zeichen je Sekunde an. Der Preis des 24-Nadlers dürfte bei 1000 Mark liegen.

Der dritte im Bunde ist der Star Laserprinter 8 (Bild). Der Laserdrucker beherrscht vier Emulationen:
— HP-Laserjet
— Epson EX-800
— Diablo 630ECS
— IBM Proprinter

Die Emulation läßt sich sogar vom Computer aus umstellen. In Kürze wird auch ein Treiber für Postscript lieferbar sein.

Star-Parade

Der Drucker wird in der Grundkonfiguration mit 1 MByte geliefert. Eine Speichererweiterung und zusätzliche Fonts sind ebenfalls erhältlich. Star legt nach eigenen Aussagen besonderen Wert auf Service. Alle Prozeßelemente sind beim Laserprinter 8 in einer einzigen auswechselbaren Kartusche untergebracht. Diese soll genau wie der Behälter für den Toner einfach gewechselt werden können. Der Preis des Laserjet beträgt 6487 Mark ohne Mehrwertsteuer. (ub)

Star Micronics Deutschland, Mergenthaleralllee 1-3, 6236 Eschborn/Ts., Tel. 06196/7018-0



Der Laserprinter 8 ist einer von Stars »Neuen«

Standfest

Okidata beabsichtigt auf der CeBIT zwei neue 24-Nadel-Drucker vorzustellen:
— den Microline 390 und das breite Modell
— Microline 391.

Beide Drucker sind für den Heavy-Duty-Bereich gedacht. Neben der komfortablen Steuerung über ein Front-Panel überzeugen die Drucker laut Herstellerangaben besonders durch ihre Robustheit. Der Listen-Preis wird bei etwa 1600 Mark liegen. In den absoluten Low-Cost-Bereich möchte Okidata nicht vorstoßen. Nach Aussage von Hubert Ernst, Direktor Marketing und Vertrieb von Okidata Deutschland, setzt das Unternehmen mehr auf das mittlere Preissegment mit entsprechender Leistung insbesondere in puncto Haltbarkeit der Druckköpfe.

Des weiteren bietet Okidata den Laserjet 6 mit einer neuen Emulation an. Er soll jetzt auch voll kompatibel zur HP-2-Serie sein. Bis zu 2,5 MByte Speicher erlauben es nun, eine volle Grafikseite einzulesen, sowie auch Makros zu laden. Der Preis des Laserdruckers beträgt 4600 Mark ohne Mehrwertsteuer. Die Speichererweiterung um 1,5 MByte kostet 950 Mark. (ub)

Okidata GmbH, Hansaallee 187, 4000 Düsseldorf 11, Tel. 0211/597940

Zeichensätze für Malprogramme

Zur Gestaltung von Grafiken, die mit gängigen Malprogrammen entwickelt werden, können auch verschiedene Schriften beitragen. Gleich 50 verschiedene Zeichensätze finden sich auf der »Headline Type-Brushes«-Diskette von Klaus Juris, die zu einem Preis von 89 Mark bei nachstehender Adresse angefordert werden können. Sie sind im IFF-Format gespeichert, so daß sie beispielsweise bei Deluxe-Paint in den zweiten Grafikscreen geladen und als Brush weiterverwendet werden können. Die einzelnen Schrifttypen (beispielsweise »Typewriter« oder »Helvetica«) sind sehr vielfältig und stellen eine Bereicherung für jeden Grafikanwender dar.

Eine zweite, ebenfalls 89 Mark teure Diskette mit dem »Chart-Construction-Set« ermöglicht eine leichte Chart-Gestaltung, ohne hierfür gleich ein Kalkulationsprogramm zu benötigen. Aus vorgefertigten Bausteinen lassen sich mit einem IFF-Zeichenprogramm Balken-, Linien-, Kuchen- und Plus/Minus-Diagramme erstellen. Der Anwender findet Blanko-Charts, vielfältige Beschriftungs- und Kennzeichnungsmöglichkeiten, verschiedene Symbole wie Landkarten oder Länderflaggen oder Kfz-Kennzeichen. In Verbindung mit den oben beschriebenen Headline Type-Brushes lassen sich vielfältige Chart-Grafiken erstellen. (dm)

Klaus Juris Grafik-Design, Bahnhofstr. 106, 6392 Neu-Anspach, Tel. 06081/8590

Speicher satt!

Für alle speicherhungrigen Amiga 2000-Besitzer ist die Rettung da: Volle 8 MByte Zusatzspeicher auf nur einer Steckkarte! Die von ASDG Inc. gefertigte RAM-Erweiterung, die in einen der 100poligen A2000-Slots eingesteckt wird und so groß wie eine 2-MByte-Karte ist, erweitert so platzsparend den Hauptspeicher. Zur 4495 Mark teuren, voll mit 1-MBit-Chips bestückten Platine liefert der Hersteller eine sehr umfangreiche, englische Anleitung und eine Diskette zur korrekten Installation des Boards. Die Karte ist sehr kompatibel und stellt auch den Laien vor keine Einbauschwierigkeiten. Sie stellt eine sinnvolle Erweiterung für chronischen Speichermangel dar und ist bei nachstehender Adresse zu beziehen. (dm)

Intelligent Memory, Basaltstr. 58, 6000 Frankfurt, Tel. 069/7071102



Commodore besuchen

Alle Amiga-Fans, die beabsichtigen, Commodore auf der CeBIT vom 16.-23. März zu besuchen, sollten folgende Daten im Kopf haben: Der Messestand befindet sich in Halle 1 im Bereich 5g8-5h1. Commodore wird jeden Tag ein Gewinnspiel veranstalten, bei dem jeweils ein komplettes PC Textsystem und 30 Bücher zu gewinnen sind. Alle Lose müssen ausgefüllt bis 17.30 Uhr am Infostand abgegeben sein. Zusätzlich wird aus allen Losen, die während der Messe eingehen, am Ende noch ein Gewinner für den Hauptpreis ermittelt. Dieser Hauptgewinn ist ein kompletter Amiga 2000.

Ob Commodore es schaffen wird auf der CeBIT ein neues Genlock für den Amiga vorzustellen, war bei Redaktionsschluß noch nicht völlig sicher. Es soll sich aber um eine Kombination aus einem Framegrabber und einem Genlock handeln. Ebenfalls noch nicht sicher ist, ob Commodore eine Hard-Disk für den Amiga 500 auf der CeBIT präsentieren wird. Es kann allerdings wieder mit einigen interessanten Neuheiten von Zweitausstellern gerechnet werden. Somit ist der Commodore-Stand wieder ein Hauptanziehungspunkt der diesjährigen CeBIT für alle, die sich über den Amiga informieren wollen. (jk)

Slimline-Zusatzlaufwerk

Für jeden Amiga-Besitzer, der sich ein Zusatzlaufwerk wünscht, ist sicher das kompatible, in ein solides Metallgehäuse eingebaute, abschaltbare 3½-Zoll-Laufwerk von Fujitsu interessant. Das in Amigabeige gehaltene Gerät besticht durch geringe Bauhöhe und wenig Geräuschentwicklung. Leider besitzt das Gerät keinen durchgeschleiften Bus, wodurch aber der geringe Preis von 289 Mark erzielt werden konnte. (dm)

Datentechnik Bittendorf, Postfach 100248, 6360 Friedberg, Tel. 06031/61950

Endlich GFA-Basic!

Rechtzeitig zur CeBIT in Hannover wird eine Version des auf dem Atari ST bewährten GFA-Basic für den Amiga herausgebracht.

Die Programmierer von GFA-Systemtechnik arbeiten zur Zeit noch auf Hochtouren an der endgültigen Fertigstellung, damit Ende Mai eine fertige Verkaufsversion über die Ladentische gehen kann. Der geplante Verkaufspreis wird bei 198 Mark liegen. Direkt auf der Messe kann man sich jedoch schon von den meisten Fähigkeiten dieses schnellen Basic-Dialektes überzeugen. Direkt am Commodore-Stand wird Georg Zweschper, der im wesentlichen für die Programmierung der Amiga-Version zuständig war, die Fähigkeiten von GFA-Basic direkt am Computer vorführen. (jk)

GFA-Systemtechnik, Heerdter Sandberg 30, 4000 Düsseldorf 11, Tel. 0211/588011
Preis: 198 Mark

Zing! in deutscher Version

Das CLI-Hilfsprogramm »Zing!« (Test in AMIGA 6/7), das mit englischer Anleitung bereits seit längerem auf dem Markt ist, ist nun auch in einer vollkommen eingedeutschten Version erhältlich. Da Anleitung und Menütexte des Programmes übersetzt wurden, ist die Arbeit damit erheblich erleichtert. Besonders praktisch ist, daß Zing! nun auch auf die PAL-Version des Amiga angepaßt wurde. Es lassen sich somit im Zing!-Hauptprogramm (ohne Kopf- und Fußzeile) maximal 26 Zeilen (gegenüber 20 in der NTSC-Version) für Dateinamen nutzen. Besonders praktisch ist ein Tip, wie mit Hilfe eines Diskettenmonitors die vorgegebenen Gerätenamen von Zing! auf die eigenen Bedürfnisse angepaßt werden können. (A. Lietz/jk)

Markt & Technik Verlag AG, Buchverlag, Hans-Pinsel-Str. 2, 8013 Haar, Tel. 089/4613-0
Preis: 198 Mark

Professionelle Seiten

Nach langem Warten ist das von Gold Disk schon für Herbst letzten Jahres angekündigte Desktop Publishing-Programm »Professional Page« ab sofort erhältlich. Die ersten Erfahrungen mit DTP auf dem Amiga sammelte diese kanadische Firma schon mit dem »Page Setter« (Test AMIGA 6/7). Professional Page (PPage) rundet die Palette der DTP-Software jetzt nach oben ab und stellt somit eine Alternative zu bisher auf anderen Computern wie beispielsweise dem Apple Macintosh bestehenden DTP-Systemen dar. Als Zusatz für absolut professionelles Arbeiten ist von Gold Disk der »Color Separator« als Veröffentlichung noch in diesem Frühjahr in Aussicht gestellt worden. Damit könnten dann Zeitungen komplett mit vierfarbigen Bildern auf Satzmaschinen wie der Linotronic entworfen werden. Bisher gibt es PPage nur mit Druckertreibern für postscript-fähige Drucker wie etwa Laser-Drucker. Wann Konvertierungen für normale Nadeldrucker erscheinen, stand bei Redaktionsschluß noch nicht fest. (jk)

Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081
Preis: 698 Mark

»Professional Page« von Gold Disk soll mit dem Zusatz »Color Separator« erst richtiges DTP bringen

Preise rutschen

Ein neues preiswertes Genlock für den Amiga bietet Intelligent Memory an. Damit können Sie Ihre Videos mit auf dem Amiga erzeugten Bildern, zum Beispiel Titeltexten, kombinieren. »AmiGen« muß nur in den RGB-Port des Amiga eingesteckt werden (Bild). Am Eingang »Video In« schließen Sie die zweite Video-Quelle an und am Ausgang einen Monitor oder Video-Recorder. Das RGB-Signal des Amiga ist ebenfalls durchgeschleift.



Das voraussichtlich 498 Mark teure Gerät erlaubt allerdings nur das Mischen des Amiga-Signals mit der externen Video-Quelle und arbeitet nur im Interlace-Modus. Die Bildqualität steht laut Anbieter der von wesentlich kostspieligeren Genlocks in nichts nach. Zur Zeit ist allerdings nur die NTSC-Version lieferbar, das PAL-Interface wird vermutlich in drei Monaten fertiggestellt werden. (ub)

Intelligent Memory, Basaltstr. 58, 6000 Frankfurt 90, Tel. 069/7071102

Universal-Tastenfeld

Eine neuartige Stand-alone-Tastatur« wurde jetzt von AFC-Technology entwickelt, die sich besonders für Kassenanwendungen, Prozeßsteuerungen und Informationssysteme eignet. Das bis zu 64 Tasten große Tastenfeld namens »Multikey« ist entweder alleine oder mit der normalen PC-Tastatur an jedem PC/AT und natürlich am Amiga 2000 zu verwenden. Es lassen sich sogar verschiedene Tastenfolgen einspeichern. Schlüsselschalter, Relaisausgang für die Schaltung externer Geräte sowie eingebaute Barcodeleser sind optional erhältlich. Der Preis liegt je nach Ausstattung zwischen 930 und 3120 Mark. (jk)

AFC-Technology, Bürgerbuschweg 48, 5090 Leverkusen 3, Tel. 02171/80057

News im Überblick

Besser und sicherer

Damit ist ein weiterer Update der Terminalsoftware charakterisiert, welche in der 68000er 9/87 einem ausführlichen Test unterzogen wurde. Mit »Online 2.1« ist kurz nach der Einführung der ersten Version schon eine von allen bekannten Fehlern befreite neue Ausgabe erschienen. Es wurden mehrere Zeichensätze integriert und auch die Terminal-Emulation VT-102 wird nun unterstützt. Die VT-Emulationen positionieren jetzt alle Zeilen richtig und stellen damit nun eine annehmbare Emulation dieser Terminals dar.

Die Möglichkeiten des Telefonbuches wurden genauso wie die einer einblendbaren Uhr erweitert. Vielfältige Bildschirmeinstellungen unterstützen jetzt auch den Interlace-Modus, der in PAL und Overscan erhebliche Datenmengen darstellbar macht.

(Ottmar Röhrig/jk)

Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399, Preis: 175 Mark

Programmsammlung in Modula-2

Meier Vogt, die den Modula-2-Compiler »M2Amiga« anbieten, bauen zur Zeit eine Programmsammlung auf Public Domain-Disketten auf, die dann zum Selbstkostenpreis erhältlich sein soll. In einem Rundschreiben an ihre Kunden haben Meier Vogt alle Modula-2-Programmierer aufgefordert, ihre »Abfallprodukte« zur Verfügung zu stellen. Diesem Aufruf möchten wir uns gerne anschließen. Da »M2Amiga« auch die Grundlage des Compilers zum Modula-2-Kurs in der AMIGA bildet, ist diese Initiative auch für AMIGA-Leser interessant: Sie können auf die fertigen Programme zurückgreifen und auch mit eigenen Routinen den Aufruf unterstützen. Wenn Sie also meinen, ein paar interessante Modula-2-Routinen zu haben — einschicken. Sie helfen allen Programmierern. Mit Hilfe einer für jedermann zugänglichen Sammlung von Programmen wird Modula-2 sicher noch mehr Freunde auf dem Amiga finden. (ub)

A.+L. Meier Vogt, Im Späten 23, CH-8906 Bonstetten/ZH, Tel. 0041-1-7003037



Die Profex-Speichererweiterung paßt sich an

Passend zum Amiga 500

Für alle Amiga 500-Besitzer, die ständig zu wenig Hauptspeicher zur Verfügung haben, bietet sich geradezu der Kauf einer Speichererweiterung an. Batavia bietet eine speziell an den Amiga 500 angepaßte Speichererweiterung an. Die Zusatzbox »Profex SE 2000«, die zum Design des Computers paßt (Bild) und einfach an den seitlichen Erweiterungsport angesteckt wird, erweitert den Hauptspeicher um 2 MByte. Sie verfügt über einen durchgeschleiften Expansionsbus für weitere Geräte. Die Erweiterung läßt sich abschalten. Bestückt ist die 898 Mark teure Box mit 64 schnellen D-RAM-Bausteinen mit 120 ns. Ebenfalls von Batavia ist ein 3½-Zoll-Zusatzlaufwerk für alle Amiga-Modelle erhältlich. Das in ein solides, abgeschirmtes Metallgehäuse eingebaute Slimline-Laufwerk verfügt über einen durchgeschleiften Bus und paßt in der Farbe zum Amiga (Bild). Der Anschluß des 329 Mark teuren Diskettenlaufwerks erfolgt problemlos durch einfaches Anstecken. (dm)

Batavia, M. Sawatzky GmbH, Niedernhart 1, 8391 Tiefenbach, Tel. 08546/190

Go (on) Amiga!

Eine der ersten Dateiverwaltungen auf dem Amiga war Softwood File. Für eine Anpassung auf deutschsprachige Verhältnisse sorgte das schweizerische Unternehmen Softwareland. Das Ergebnis war GoAmiga! Datei. Eine einfach zu bedienende Dateiverwaltung. Nun hat Softwood »Write & File« herausgebracht. Softwareland zieht mit »GoAmiga! Text« nach. Und das ist eine Textverabeitung inklusive einer integrierten — allerdings abgespeckten Version — von GoAmiga! Datei.

Nicht übernommen wurde die Verwaltung von Grafik- oder Sound-IFF-Dateien. Die Textkomponente soll eine auf drei Lexika basierende Rechtschreibkorrektur enthalten. Damit lassen sich ein Hauptlexikon und mehrere Fachlexika anlegen. Ohne die Fähigkeit der Einbindung von Grafiken in den Text wendet sich das Programm an WYSIWYG gewöhnte Zeitgenossen. Serienbrieffunktionen sowie Kopf- und Fußzeilen sollen ebenfalls vorgesehen sein. Als Schnittstelle zwischen Datei und Text fungiert das Clipboard. Texte lassen sich so aus einem Dokument ausschneiden, in einen Datensatz einfügen und umgekehrt. Kleine Details unterstützen das Konzept der Entwickler, einen für Computeranfänger leichten Einstieg in die Text-/Datenverarbeitung zu schaffen. GoAmiga! Text soll für 299 Mark ab April erhältlich sein. (pa)

Softwareland, Franklinstr. 27, CH-8050 Zürich, 0041/1-3115959
Vertrieb BRD: Amiga Art Machine, Planegger Str. 1, 8000 München 60, Tel. 089/8340591

GOMF 2.0

Wer hat das noch nicht erlebt: Das gerade bearbeitete Programm steckt in den letzten Zügen, doch da stürzt der Amiga ab und meldet eine »Guru Meditation«. Der kanadische Programmierer Christian Johnsen muß in diesem Fall mit dem Wutschrei »Get Outa My Face!« reagiert haben, denn sein Programm zum Abfangen von Amiga-Abstürzen wurde nach diesem Schlachtruf »GOMF« genannt. Ist »GOMF« einmal geladen, belegt er 21,8 KByte Speicher des Amiga. Sobald der Amiga das »Software Error Fenster auf den Bildschirm bringt, würde normalerweise nach einem Klick auf »CANCEL« die bekannte »Guru Meditation« erscheinen. Statt dessen erscheint nun ein Fenster von GOMF, mit dessen Hilfe die Requester und Screens des Programmes, das den Fehler auslöste, gelöscht werden können.

Ob GOMF wirklich alle Guru Meditations beseitigen kann, erfahren Sie im Test in einer der nächsten Ausgaben.

(Andreas Lietz/jk)

Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081, Preis: etwa 100 Mark

Amigos für DEHOCA

Beim Dachverband der deutschen Computeranwender DEHOCA findet zunehmend der Amiga Anklang. Nicht nur, daß unter den zahlreichen Mitgliedern viele Amiga-Freunde an den Tastaturen sitzen, auch bei einem Wettbewerb für die grafische Gestaltung eines Titelbildes für die DEHOCA siegte eindeutig ein mit Deluxe Paint erstelltes Motiv. Auf der CeBIT wird der DEHOCA im Rahmen des Computercamps wieder aktiv sein. Der Monatsbeitrag beträgt 8 Mark beziehungsweise 5 Mark für Auszubildende. Wesentliche Neuerung ist der Print-Public-Pool, der günstige Anschaffungen an Computerzubehör möglich macht und dessen Telefon-Hotline unter der Nummer 05722/26939 für alle Mitglieder erreichbar ist. Weitere Informationen gibt es auch in der DEHOCA-Mailbox unter 05722/3848 mit 7e1. (jk)

DEHOCA, Rintelner Str. 15, 3062 Bückeburg

Scribble: mit Umlauten

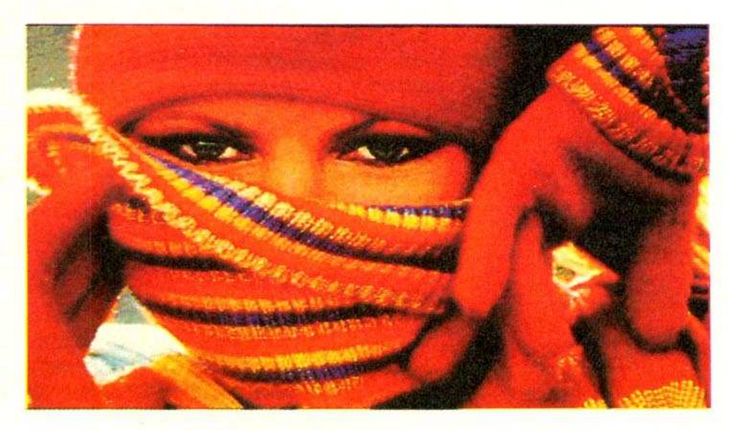
Die Textverarbeitung »Scribble« ist ab sofort in der neuesten Version 2.10 in dem Software-Paket »The Works« integriert. Darin sind zusätzlich noch »Analyze« und »Organize« vom gleichen Hersteller enthalten. Scribble bietet außer dem jeweils eingestellten Zeichensatz auch noch alle über die ALT-Tasten erreichbaren Zeichen sowie die Unterstützung des PAL-Modus. Das heißt, daß Scribble nun direkt nach dem Laden mit einem großen Fenster vorliegt und auch das seitenweise Vor- beziehungsweise Rückwärtsrollen im Text korrekt funktioniert. Weiterhin wird die Datei mit den Standardeinstellungen (Scribble.fmt) nicht mehr auf der Boot-Diskette, sondern auf der Disk, auf der sich das Programm befand, gesucht. So müssen Sie Ihre Konfigurationsdatei nicht mehr auf die Workbench kopieren, wollen Sie Scribble nicht booten.

Auch das Texthandling wurde wesentlich verbessert. So werden jetzt zum Beispiel Zeilen nach einem Sprung an deren Ende nicht mehr mit Leerzeichen aufgefüllt. Dies geschieht erst, nachdem der Cursor die Zeile verlassen hat, was der Übersichtlichkeit sehr zuträglich ist. Neben einem schnell geladenen Verzeichnis fällt angenehm auf, daß im Fenster für Diskettenoperationen nun auch Unterverzeichnisse angezeigt werden, in die Sie per Mausklick »einsteigen« können.

Leider wurde jedoch vergessen, die Suchfunktion dahingehend zu verbessern, daß diese nicht immer automatisch ab dem Anfang des Textes mit der Suche beginnt. Das ist vor allem beim Durchsuchen eines Textes nach Stichwörtern hinderlich.

Sämtliche Druckerbefehle sind jedoch gleichgeblieben, so daß eine volle Kompatibilität zu alten Dateien gewährleistet ist. Die gesamte Benutzeroberfläche wurde in Bedienung und Aussehen der neuen Reihe von MicroSystems-Software angepaßt. Das ist für viele sicherlich ein Anreiz, da jetzt eine Gruppe von leistungsfähigen Programmen zur Verfügung steht, die auf schon bekannten Grundfunktionen aufbauen und als Gesamtpaket erhältlich sind. (O. Röhrig/jk)

Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081
Preis The Works: 400 Mark



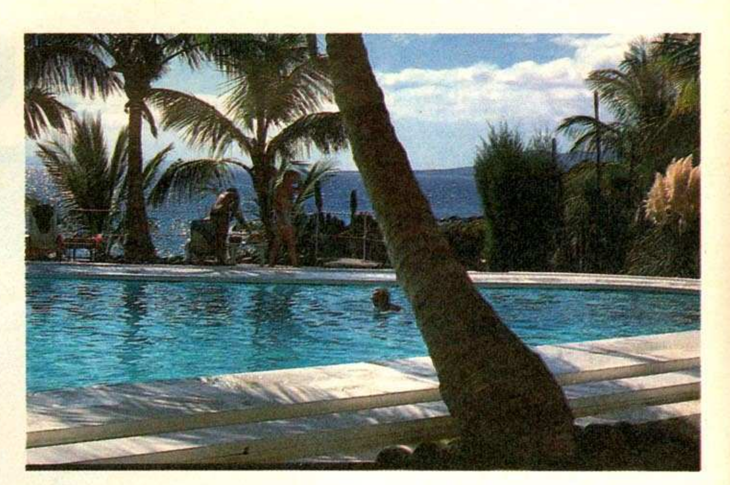
»Digi-View« läßt sich mit Farbscheibe einfacher bedienen

Digi-View rotiert

Für den weitverbreiteten Videodigitizer Digi-View bietet ein Münchner Software Shop einen Nachrüstsatz an, der es erlaubt, das Farbrad automatisch zu drehen. Die für 148 Mark erhältliche Erweiterung zu Digi-View 2.0 besteht laut Produktinformation aus einem Motor, einem Anschlußkabel und einer deutschen Umbauanleitung.

Der Motor mit der Farbscheibe wird mit dem zum Digitizer gelieferten Haltebügel an der Kamera befestigt. Der Anschluß des Motors erfolgt über den freien Joystickport. Die Steuerung des Motors wird durch Digi-View 2.0 bereits unterstützt. Ist der Motor installiert, erscheinen automatisch zwei neue Menüpunkte. Mit »Auto« soll es dann ohne weiteres Zutun möglich sein, ein Farbbild zu digitalisieren. (ub)

Musik- und Grafiksoftware-Shop, Wasserburger Landstr. 244., 8000 München 82, Tel. 089/4306207
Preis: 148 Mark



Auf Motivsuche: für Amiga-Videos auf Weltreise

Hochseetauglich

Ganz im Zeichen von Desktop Video auf dem Amiga steht eine Kreuzfahrt mit der »Maxim Gorki«. Für Videoaktive, die im Urlaub nicht von ihrem Hobby lassen können oder Video und Amiga erst einmal richtig kennenlernen möchten, hat sich die NUR-Seereisen etwas Besonderes einfallen lassen: Seit mehr als zwei Jahren organisiert »Art Basic Audio, Langenfeld« Videoseminare auf der Maxim Gorki und richtet zu diesem Zweck ein komplettes Videostudio an Bord des Liners ein. Den letzten Schliff erhält das Bordstudio laut Aussage des Organisators durch die Integration der Amiga-Hard- und Software.

Reiner Folgner von Art Basic Audio meint dazu: »Wir wissen, daß viele Amiga-Besitzer durch den Einsatz von Peripheriegeräten wie Genlock oder Digitizer ein reges Interesse am Medium Video zeigen. Es erlaubt ihnen zum Beispiel, die mit Animationsprogrammen entworfenen Bildsequenzen statt auf Diskette direkt auf Videoband zu überspielen. Bei unseren Seminaren stehen attraktive Videogeräte, Camcorder und Bearbeitungsschnittplätze bereit, um Video und Amiga als Hobby oder mit professionellen Ansprüchen zu verbinden. Die Ausrüstung entspricht dem neuesten Stand der Technik und stammt von Panasonic, GSE, Sony, Sennheiser, Tascam und Erno.«

Der Start der ersten 14tägigen Reise ist laut Reiseinfo für den 24. September 1988 geplant. Das Abenteuer beginnt in Genua und führt in selten besuchte Regionen Westafrikas. Am 7. Januar 1989 geht es weiter. Vier Monate wird die Maxim Gorki auf Fahrt rund um die Welt sein. Wer nicht die gesamte Strecke mitfahren möchte, bucht eine der fünf Teilstrecken, die zwischen 20 und 28 Tage dauern werden. Die Reiseroute wird von der Karibik über Australien bis zu den Malediven reichen. Die Motivsuche (Bild) fällt den Urlaubern auf der Reise sicher leicht. (ub)

Art Basic Audio, Langforterstr. 28, 4018 Langenfeld, Tel. 02173/76741
Preise: ab 3690 Mark

Kein König

Was zunächst für viele Software-Käufer wie ein schlechter Scherz klingen sollte, war in Wirklichkeit volle Absicht. Bei den ersten Versionen des Spiels »King of Chicago« von Cinemaware, die in Deutschland verkauft wurden, ging beim Booten der Hauptdiskette rein gar nichts. Was selbst findige Software-Importeure nicht ahnen konnten, war, daß diese Versionen des Spiels nur für den amerikanischen Markt gedacht sind. Das Programm fragt beim Hochfahren ab, was für ein Amiga vorliegt, und zwar in bezug auf PAL- oder NTSC-Norm. Durch diese Abfrage soll der europäische, speziell der deutsche Markt von Grauimporten aus den Staaten freigehalten werden. Es bleibt abzuwarten, ob auch andere Firmen zu dieser Methode greifen. Man müßte sich in Zukunft auf etwas Wartezeit einrichten, bis von den Programmen auch jeweils eine Version verfügbar ist, die auf PAL-Amigas überhaupt läuft. Damit ist dann natürlich noch nicht die PAL-Version mit entsprechend mehr Bildschirmzeilen gemeint. (jk)

Schnelle Treiber

»Turbo-Print« ist ein Utility, das Grafikausdrucke beschleunigt. Laut Aussage des Herstellers Irsee-Soft unterstützt das Programm alle gängigen Drucker für den Amiga. Turbo-Print wird nach dem Einschalten des Amiga geladen und steht dann resetfest zur Verfügung. In einem speziellen Menü können alle Parameter zum Drucken spezifiziert werden. Sogar H.A.M.-Bilder sollen sich auf Farbdruckern schnell ausgeben lassen. In einer unserer nächsten Ausgaben werden wir das Programm unter die Lupe nehmen. (ub)

Irsee-Soft, Grüntenstr. 6, 8951 Irsee, Tel. 08341/8211
Preis: 89 Mark

BBS-PC wird benutzerfreundlich

Das Mailbox-Programm BBS-PC von MicroSystems wird nun in einer neuen Version vertrieben. Sie trägt die Bezeichnung 4.2 und hat viele Neuerungen, von denen hier nur die interessantesten angesprochen sein sollen.

Neben dem glücklicherweise schon zu Standardausrüstung eines Updates gehörenden PAL-Bildschirm hat sich vor allem auf der Bedienerseite vieles getan. Es werden nun sowohl Modems mit 4800 als auch mit 9600 Bit/s unterstützt. Leider sind einige Definitionsdateien aber nicht mehr kompatibel zu den Äquivalenten der früheren Versionen. Dazu bekommen Sie aber Konvertierungsprogramme geliefert, so daß sich hier kein eigentliches Problem stellt. Bei den höheren Übertragungsraten ist es nur konsequent, daß auch andere Protokolle angeboten werden. So sind nun neben den gängigeren Protokollen auch YModem (zusätzlich in der Batch-Version), ZModem und Kermit vorhanden. Mit der Batch-Version von YModem ist der Benutzer in der Lage, sich mehrere Dateien hintereinander zu überspielen. Damit entfällt nicht nur die lästige Unterbrechung bei jeder neu zu übertragenden Datei, sondern es werden auch die Verbindungszeit und damit die Kosten erheblich herabgesetzt.

Auffällig ist die automatische Verwendung von Farben in den Menüs. Diese können Sie allerdings nur mittels einer ANSI- beziehungsweise VT102-Emulation ausnutzen. Bei anderen Terminals werden die Codes als ASCII-Zeichen angezeigt. Eine Möglichkeit, dieses abzustellen, gibt es jedoch nicht. Weiterhin können Sie als Benutzer jetzt mit dem Semikolon mehrere Kommandos direkt hintereinander ausführen. Das spart Zeit und erlaubt das Schreiben einfacherer Batch-Dateien.

BBSFIX und BBSINIT wurden in einigen Punkten erheblich verbessert, so daß die Box nun nach einem Absturz theoretisch alle wichtigen Dateien selbst reparieren und damit den Betrieb wieder aufnehmen kann.

Erfreulicherweise wurden alle zeitkritischen Routinen so umgeschrieben, daß Sie ohne Bedenken BBS-PC auch mit einem 68020- oder einem 68010-Prozessor benutzen können. Dies ergab in früheren Versionen des öfteren Probleme. Mit der neuen Benutzerführung sind jetzt auch Menüzeilen mit mehr als 80 Zeichen möglich. Damit lassen sich viele Menüpunkte in wenigen Zeilen unterbringen, was Bildschirmplatz spart. Das ist vor allem bei größeren Terminal-Emulationen von besonderer Bedeutung. Eine spezielle Funktion erlaubt das Einblenden verschiedenster Zeiten innerhalb eines Menüs, so daß der Benutzer beispielsweise immer über die noch verbleibende Zeit informiert ist. Ähnlich dem Begrüßungstext ist nun auch ein Abschlußtext vorhanden. Hier können zum Beispiel Telefonkosten berechnet und eingeblendet werden, was dem Benutzer sicherlich oft weiterhilft.

Dem allgemeinen Trend bei Neuerscheinungen folgend wurde die Multitaskingfähigkeit von BBS-PC erhöht. Nun kommt der Bildschirm beispielsweise nicht immer sofort nach vorne, wenn sich ein Anrufer »einloggt« und die Daten werden von der Diskette mit einem großen Puffer gelesen, was die Ladezeiten erheblich herabsetzt und ein »Hin- und Herkratzen« auf der Diskette weitestgehend unterbindet. Somit ist das Programm sicherlich nun noch eine bessere Wahl für Mailbox-Enthusiasten, die nicht die Zeit für das Programmieren einer eigenen Box haben. (Ottmar Röhrig/jk)

Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399
Preis: 175 Mark

Der Schritt zur einfachen Grafik



»The Graphics Studio« verheißt einen neuen Grafikkomfort

Das brandneue Mal- und Zeichenprogramm für den Amiga kommt vom renommierten Spielehersteller Accolade aus Amerika. »The Graphics Studio« ist der erste wirklich ernstzunehmende Konkurrent für Deluxe-Paint, zumal es nur ungefähr halb soviel kostet. Neben den üblichen Funktionen wie Rechtecke, Ellipsen und Polygone bietet The Graphics Studio natürlich auch alle Feinheiten beim Arbeiten mit Brushes, wie etwa Größenänderungen (Bild). Die Menübedienung wurde neu definiert und hält sich damit nicht an den Intuition-üblichen Standard. Dies erfordert eine Einarbeitungszeit, hat aber auch Vorteile. The Graphics Studio arbeitet nur in Lores mit maximal 32 Farben und wird in Deutschland zirka 130 Mark teuer sein. Wir werden das Programm einem ausführlichen Test unterziehen, den Sie in einer der nächsten AMIGA-Ausgaben lesen können. (jk)

Intelligent Memory, Basaltstr. 58, 6000 Frankfurt, Tel. 069/7071102
Preis: 129 Mark



In der **Ausgabe 4/88** wagt Happy-Computer einen Blick in die Zukunft. Wie sieht unser Leben mit dem Computer aus? Welche Chancen bietet er? Welche Gefahren schlummern in ihm? In Science-fiction-Geschichte schreiben Leser, was sie von der Zukunft erwarten. Lassen Sie sich von vielen tollen Ideen überraschen.

Der Amiga ist ein Grafik-Könner ohnegleichen. Doch wer kann schon malen wie ein echter Künstler? Ein gelernter Grafiker zeigt, daß das Malen von tollen Bildern gar nicht so schwer ist. Er verrät seine Tricks, damit Sie die Fähigkeiten des Amiga voll ausnutzen können. Für alle Grafik-Fans gibt es einen riesigen Malwettbewerb, bei dem es viel Geld zu gewinnen gibt. Vielleicht kommen Sie oder ein anderer Amiga-Künstler auf die ersten Plätze.

Die **Ausgabe 5/88** bringt Licht in die Geschichte des Amiga. Wie fing alles an? Wer hat ihn entwickelt. Wieso kaufte Commodore Amiga? Welche Rolle spielte dabei Atari? Wie wurde das Videospiel zum Traumcomputer? Die Amiga-Story ist spannend wie ein guter Krimi und hält auch für Sie einige Neuigkeiten parat, die bisher noch niemand wußte.

Viren auf dem Amiga sind nicht nur lästig, sondern auch gefährlich. Happy-Computer erklärt, wie Computer-Viren funktionieren, welche Tricks sie benutzen, um sich unerkannt zu verbreiten, wie man sie entdeckt und wie man sich vor ihnen schützt.

Ein weiteres Thema ist Computer-Kriminalität. Geschickte Verbrecher verwenden den Computer als Werkzeug, um sich ohne Fingerabdrücke und Beweise Millionen zu ergaunern. Doch auch die Polizei nutzt Computer für die Verbrechensbekämpfung. Wie steht es in diesem Wettlauf?

Für Videoprofis: die neue Version von Videoscape

In der brandneuen Version 1.10 von Videoscape wird neben einigen anderen Funktionen nun auch der PAL-Modus voll unterstützt. Sogar der Overscan funktioniert bestens, so daß einer professionellen Videoanwendung mit dem Amiga nichts mehr im Wege steht. Machen Sie es den Designern der englischen Computer-Puppe Max Headroom nach und entwickeln auch Sie fernsehreife Hinter- und Vordergründe mit dem Amiga. Neben diesem Hauptpunkt wurden natürlich noch andere Verbesserungen eingeführt. So wird bei der Speicherung der ANIM-Dateien ein neuer Packalgorithmus verwendet, der Platzersparnisse von 50 Prozent und mehr verspricht. Dadurch können Sie nun doppelt so lange Animationen auf eine Diskette speichern. Im Hauptfenster von Videoscape werden einmal eingestellte Optionen nur auf einen bestimmten Befehl hin und nicht mehr nach jedem Durchlauf gelöscht. Damit wird das Herumprobieren wesentlich vereinfacht. Sie können nun, statt bisher einer, bis zu vier Lichtquellen vollkommen unabhängig voneinander einsetzen, um interessantere Effekte auf den Objekten zu erzielen. So ist es auch möglich, die komplexesten Szenen günstig auszuleuchten. Mit der Anzahl der gesetzten Lichtquellen steigt allerdings auch die Rechenzeit für jedes Bild rapide an. Die Programmierer bedachten auch die Tatsache, daß der Amiga ein Multitasking-Computer ist und ermöglichen dieser Version den reibungslosen Betrieb gleichzeitig mit anderen Programmen. So kann der Ton jetzt weggeschaltet und der Videoscape-Screen endlich auf- und abgeschoben werden. Auch der versehentliche Aufruf von Menüs anderer Programme ist jetzt unterbunden.

Zu einigen interessanten Effekten kommt man durch den Austausch der Bewegungsdateien von Objekten und Kamera. Dies ist jetzt auch ungehindert möglich, da beide das gleiche Format haben.

Durch die Verwendung der vier neuen Farben, die durch Mischung anderer Farben entstehen und auf die noch unbelegten Farbcodes gelegt wurden, entstehen vollkommen neue Objekte. Dieser Effekt wird durch das neue Attribut »Halb-Transparent« verstärkt. Hierbei lassen so markierte Flächen den Hintergrund durchschimmern.

In dem Hilfsprogramm »EGG« wurde der Fehler beseitigt, der manchmal Sterne ungewollt unter den Horizont setzte. Bei »OCT« ist es jetzt möglich, Objekte um eine Achse zu drehen, ohne die Reihenfolge der Verbindungspunkte- und linien zu ändern. Außerdem verarbeitet OCT nun das 3DG1-IFF-Format. Hierbei werden die Koordinaten nicht mehr in Text-, sondern in Binärdateien gespeichert, was die Diskettenoperationen auf ein Minimum verkürzt.

Inwieweit OCT jedoch nach den erschienenen Programmen »Interchange« und »FOT«, die unter anderem die Konvertierung von »Sculpt-3D«-Dateien in das Videoscape-Format erlauben, noch sinnvoll ist, muß jeder für sich entscheiden. (Ottmar Röhrig/jk)

Markt & Technik, Buchverlag, Hans-Pinsel-Str. 2, 8013 Haar, Tel. 089/46 13-0
Preis: 385 Mark

IMPRESSUM

Herausgeber: Carl-Franz von Quadt, Otmar Weber

Geschäftsführender Chefredakteur: Michael Scharfenberger

Chefredakteur: Albert Absmeier (aa)
Redaktion: Peter Aurich (pa), René Beaupoil (rb), Ulrich Brieden (ub), Jörg Kähler (jk), Dieter Mayer (dm)

Redaktionsassistenz: Cathy Winter (414)

Fotografie: Jens Jancke, Claudia Kränzle

Titelgestaltung: Heinz Rauner, Grafik Design, Werner Nienstedt

Layout: Friedemann Porscha (Ltg.), Dagmar Berninger, Willi Gründl

Auslandsrepräsentation:
Schweiz: Markt & Technik Vertriebs AG, Kollerstr. 3, CH-6300 Zug, Tel. 042-41 56 56, Telex: 862329 mut ch
USA: M & T Publishing, Inc: 501 Galveston Drive, Redwood City, CA 94063; Tel. (415) 366-3600, Telex 752-351

Manuskripteinsendungen: Manuskripte und Programmlistings werden gerne von der Redaktion angenommen. Sie müssen frei sein von Rechten Dritter. Sollten sie auch an anderer Stelle zur Veröffentlichung oder gewerblichen Nutzung angeboten werden, so muß dies angegeben werden. Mit der Einsendung von Manuskripten und Listings gibt der Verfasser die Zustimmung zum Abdruck in von der Markt & Technik Verlag AG herausgegebenen Publikationen und zur Vervielfältigung der Programmlistings auf Datenträger. Mit der Einsendung von Bauanleitungen gibt der Einsender die Zustimmung zum Abdruck in von Markt & Technik Verlag AG verlegten Publikationen und dazu, daß Markt & Technik Verlag Geräte und Bauteile nach der Bauanleitung herstellen läßt und vertreibt oder durch Dritte vertreiben läßt. Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte und Listings wird keine Haftung übernommen.

Produktionsleitung: Klaus Buck

Gesamtanzeigen-Verkaufsleitung: Ralph Peter Rauchfuss (126)

Anzeigenverkaufsleitung: Alexander Narings (780)

Anzeigenleitung: Alicia Clees (313)

Anzeigenverkauf: Christine Pfäffinger (781)

Anzeigenverwaltung und Disposition: Patricia Schiede (172), Monika Burseg (147)

Anzeigenformate: 1/1-Seite ist 266 Millimeter hoch und 185 Millimeter breit (3 Spalten à 58 mm oder 4 Spalten à 43 Millimeter). Vollformat 297 x 210 Millimeter. Beilagen und Beihefter siehe Anzeigenpreisliste.

Anzeigenpreise: Es gilt die Anzeigenpreisliste Nr. 2 vom 1. Januar 1988
Anzeigengrundpreise: 1/1 Seite sw: DM 4900,—. Farbzuschlag: erste und zweite Zusatzfarbe aus Europaskala je DM 800,—. Vierfarbzuschlag DM 2200,—.
Anzeigen in der Fundgrube: Private Kleinanzeigen mit maximal 4 Zeilen Text DM 5,— je Anzeige.
Gewerbliche Kleinanzeigen: DM 12,— je Zeile Text.
Auf alle Anzeigenpreise wird die gesetzliche MwSt. jeweils zugerechnet.

Anzeigen-Auslandsvertretungen:
England: F. A. Smyth & Associates Limited 23a, Aylmer Parade, London, N2 OPQ, Telefon: 00 44/1/3 40 50 58, Telefax: 00 44/1/3 41 96 02
Taiwan: Third Wave Publishing Corp. 1 — 4 Fl. 977 Min Shen E. Road, Taipei 10581, Taiwan, R.O.C., Telefon: 0 08 86/2/7 63 00 52, Telefax: 0 08 86/2/7 65 87 67, Telex: 078 529 335

Vertriebsleiter: Helmut Grünfeldt (189)

Vertrieb Handelsauflage: Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz: Pegasus Buch- und Zeitschriften-Vertriebsgesellschaft mbH, Hauptstätterstraße 96, 7000 Stuttgart 1, Telefon (07 11) 64 83-0

Bezugsmöglichkeiten: Leser-Service: Telefon 0 89/46 13-2 49. Bestellungen nimmt der Verlag oder jede Buchhandlung entgegen. Das Abonnement kann jederzeit zum Ende des bezahlten Zeitraums gekündigt werden.

Bezugspreise: Das Einzelheft kostet DM 7,—. Der Abonnementspreis beträgt im Inland DM 79,— pro Jahr für 12 Ausgaben. Darin enthalten sind die gesetzliche Mehrwertsteuer und die Zustellgebühren. Der Abonnementspreis erhöht sich um DM 18,— für die Zustellung im Ausland, für die Luftpostzustellung in Ländergruppe 1 (z. B. USA) um DM 38,—, in Ländergruppe 2 (z. B. Hongkong) um DM 50,—, in Ländergruppe 3 (z. B. Australien) um DM 68,—.

Druck: R. Oldenbourg GmbH, Hürderstr. 4, 8011 Kirchheim

Verantwortlich: Für redaktionellen Teil: Albert Absmeier. Für Anzeigen: Alicia Clees.

Redaktions-Direktor: Michael M. Pauly

Vorstand: Carl-Franz von Quadt, Otmar Weber

Anschrift für Verlag, Redaktion, Vertrieb, Anzeigenverwaltung und alle Verantwortlichen:
Markt & Technik Verlag Aktiengesellschaft, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon 0 89/46 13-0, Telex 5 22 052

Telefon-Durchwahl im Verlag:
Wählen Sie direkt: Per Durchwahl erreichen Sie alle Abteilungen direkt. Sie wählen 0 89-46 13 und dann die Nummer, die in Klammern hinter dem jeweiligen Namen angegeben ist.

P 2200 – DAS PREIS-LEISTUNGS-GENIE

Die Computer-Anwender haben Grund zum Jubeln!

Genial – endlich ein Drucker, der für Einsteiger, Aufsteiger und Semiprofis geeignet und vor allem erschwinglich ist. Denn NEC erschließt Ihnen jetzt die modernste 24-Nadel-Drucktechnologie zu einem erstaunlich günstigen Preis.



Vielseitige, durchdachte Papierzuführungen.

NEC Pinwriter P 2200 mit automatischer Einzelblattführung

Erleben Sie eine neue Dimension: gestochen scharfen Korrespondenzdruck mit ungewöhnlich reicher Schriftartenauswahl, brillante Grafik-Darstellung, bequeme Druckersteuerung und integrierte Papierzuführungen.

Warum also tief in die Taschen greifen, wenn es schon für wenig Geld 24-Nadel-Technologie mit allen Raffinessen gibt?

PROFIQUALITÄT ZUM AMATEURPREIS

NEC ist mit seinen 24-Nadel-Druckern in Deutschland marktführend.

Was den P 2200 als echten Profi auszeichnet, sind seine hohe Auflösung von 360 x 360 dpi, ein halbes Dutzend serienmäßiger Schriftarten und eine Reihe praktischer Papierzuführungen. Zum Beispiel können Sie zwischendurch einen Brief drucken, ohne daß das Endlospapier extra herausgenommen werden muß.

Durch seine volle Kompatibilität mit den NEC Pinwritern der 24-Nadel-Serie harmoniert der P 2200 mit allen wichtigen Software-Paketen.

EIN NEC DRUCKER FÜR JEDERMANN

Endlich braucht niemand mehr auf die bewährte NEC Produkt- und Druckqualität zu verzichten.

DAS ELEKTRISCHE FELD

Hervorragende Druckqualität durch bewährte 24-Nadel-Technologie.

Eine breitgefächerte Gruppe – vom Schüler über den Heimanwender bis hin zum Freiberufler – findet im P 2200 die ideale Drucklösung. Anwendern, die schon seit langem auf der Suche nach einem preisgünstigen Drucker für ihren Computer sind, eröffnet er die Möglichkeit, Druckergebnisse in bewährter NEC-Qualität zu erzielen.



12 Schriftartenkassetten zusätzlich erhältlich.

Damit ist der P 2200 die optimale wirtschaftliche und technische Alternative für alle, die sich bei gleichem finanziellen Aufwand bisher nur mit antiquierten 9-Nadel-Druckern begnügen mußten.

Für Beratung und Service stehen Ihnen Ihr NEC Drucker-Fachhändler und sein Team zur Verfügung.



Weitere Informationen erhalten Sie von:

NEC Deutschland GmbH
Klausenburger Straße 4, 8000 München 80
Tel.: 0 89/9 30 06-0, Telefax: 0 89/93 77 76/8
Telex: 5 218 073 und 5 218 074 nec m

Das 1x1

Computer sind zwar dumm, dafür aber enorm fleißig. Richtige Eigenschaften für die Verwaltung großer Datenmengen.

Wissen Sie, was eine »sequentielle« Datei ist? Selbst wenn Sie diese Frage noch halbwegs richtig beantworten können, wie steht es mit den Begriffen »relative« oder »index-sequentielle« Dateiorganisation? Ob Sie nun als engagierter Anwender oder als Programmierer wissen wollen, wie Dateiverwaltung eigentlich funktioniert — wir bringen Licht in den Dschungel der Fachbegriffe. Damit den Praktikern unter Ihnen vor lauter Theorie das Thema nicht zu langweilig wird, erläutern wir die Verfahren anhand kurzer Beispielprogramme. Mit grundlegenden Programmierkenntnissen sind Sie schnell in der Lage, diese nachzuvollziehen und gegebenenfalls zu erweitern.

Dateien, Datensätze und Felder

Obwohl die Beispielprogramme in Basic entwickelt sind, sollten sich C- oder Pascal-Programmierer nicht gleich uninteressiert abwenden. Beide Demoprogramme sollen den prinzipiellen Ablauf darstellen und bestehen nur aus wenigen Zeilen. Die verwendete Technik ist unabhängig von der Programmiersprache. Zum Beispiel wird in Basic zum »Öffnen einer Datei« die Anweisung OPEN verwendet, in C die Anweisung FOPEN und in Pascal schließlich ASSIGN (und REWRITE oder RESET). Trotz der unterschiedlichen Bezeichnungen — das Ergebnis ist in allen Fällen identisch.

Dateien werden üblicherweise nach der Art der darin gespeicherten Daten unterschieden. Zum Beispiel enthält eine Programmdatei eine Folge von Programmcodes, eine Textdatei eine Folge von Zeichen und eine »Datendatei« eine Folge von Adressen, Artikeln oder ähnlichen Daten.

Uns interessiert vorwiegend der letzte Typ — die »Datendatei«. Dieser Dateityp kann sehr gut mit einem herkömmlichen Karteikasten verglichen werden. Ebenso wie ein Karteikasten enthält eine Datendatei »gleichartige« Daten. Das können beispielsweise Adressen oder Artikel sein, in der Regel jedoch nicht beides zusammen. In der Praxis ist es sinnvoller, für unterschiedliche Datensorten mehrere Karteikästen beziehungsweise mehrere »Dateien« anzulegen.

Wie ein Karteikasten enthält auch eine Datei »Karteikarten«, die sogenannten »Datensätze«. Ein Datensatz besteht aus verschiedenen Komponenten, den »Datensatzfeldern« (Bild 1). Nehmen wir ein Beispiel: Sie wollen 100 Adressen in einer Datei speichern. Dann entspricht jede Adresse genau einem Datensatz und jeder Teil einer Adresse einem Datensatzfeld. Möglicherweise setzen sich Ihre Adressen aus den Komponenten Name, Vorname, Straße, Postleitzahl, Wohnort zusammen. Die Adreßdatei besteht somit aus einer Folge von 100 Datensätzen mit jeweils fünf Feldern.

Dateien besitzen »logische« und »physische« Aspekte. Unter dem logischen Aspekt ist die Strukturierung der Datei, also die Unterteilung in Datensätze und Datensatzfelder, zu verstehen. Der andere Aspekt beschreibt, wo und in welcher Form sich die Daten physisch, also körperlich, tatsächlich auf dem Datenträger befinden und wie darauf zugegriffen wird. Die Aufgabe des Programmierers ist es, mit seinem Programm eine geeignete »Zugriffsmethode« zu wählen. Prinzipiell stehen ihm — unabhängig von der verwendeten Programmiersprache — zwei Möglichkeiten zur Verfügung:
— der sequentielle Zugriff,
— der relative Zugriff oder Direktzugriff

Je nach verwendeter Zugriffsart spricht man auch von »sequentiellen Dateien« oder »Direktzugriffsdateien«. Die verwendete Dateiform bestimmt wesentlich das praktische Verhalten einer Dateiverwaltung, die Geschwindigkeit bei der Suche oder Ändern von Daten sowie die Anzahl der verfügbaren Daten.

»Sequentiell« bedeutet: der Reihe nach oder hintereinander. Beim sequentiellen Zugriff können Sie nur der Reihe nach auf Ihre Datensätze zugreifen. Es ist nicht möglich, einen bestimmten Datensatz mitten aus der Datei heraus zu lesen.

Der Grund dafür wird klar, wenn man sich vor Augen hält, wie die Informationen auf der Diskette gespeichert sind. Bild 2 zeigt eine sequentielle Datei, die als ersten Datensatz die Adresse von Herrn »Maier« und als zweiten die von Herrn »Müller« enthält. Adressen wie die der Herrn »Maier« und »Müller« besitzen eine für Ihren Computer höchst unangenehme Eigenschaft: ihre Länge ist variabel. Da jede Adresse eine andere Länge besitzt, kann der Amiga nicht berechnen, wo sich die Adresse Nummer X befindet. Der Computer kennt bei sequentiellen Dateien nur die Position des Dateianfangs auf der Diskette und die Gesamtlänge der Datei.

Bild 1. Von der Karteikarte zur modernen Dateiverwaltung mit dem Computer

Das Einlesen von Daten aus sequentiellen Dateien beginnt immer am Dateianfang. Mit einer je nach Programmiersprache unterschiedlichen Anweisung kann genau ein Feld eines Satzes in den Hauptspeicher des Computers gelesen

Datensätze auf der Diskette

werden. Um zum Beispiel die zweite Adresse zu lesen, müssen Sie diese Anweisung fünf mal wiederholen, um zunächst alle fünf Felder der ersten Adresse zu (über)lesen. Nun befindet sich der Schreib-/Lesekopf des Diskettenlaufwerks

der Dateiverwaltung

am Beginn der zweiten Adresse. Die sechste Leseanweisung liest das erste Feld der gesuchten Adresse, den Namen »Müller«, ein. Der nächste Lesezugriff liest den Vornamen der zweiten Adresse ein und so weiter.

Sequentielle Dateien

Wollen Sie die Adresse Nummer 80 ändern, ist der gleiche Aufwand nötig. Ausgehend vom Dateianfang lesen Sie Feld für Feld der vorhergehenden 79 Datensätze ein. Nicht, weil Sie diese Felder interessieren würden, sondern nur, um den Schreib-/Lesekopf zur gewünschten Adresse zu bewegen. Anschließend können Sie mit den Schreibanweisungen der von Ihnen verwendeten Sprache die neue Adresse auf die Diskette ablegen

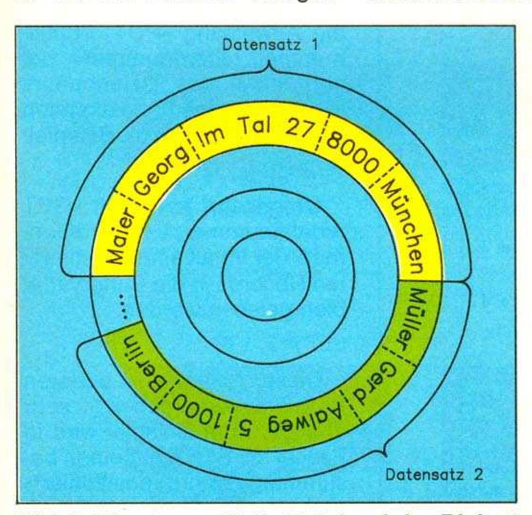


Bild 2. Die sequentielle Datei auf der Diskette

und damit den alten Datensatz überschreiben. Aber was passiert, wenn die neue Adresse länger ist als die alte und daher mehr Platz benötigt? Mit den Schreibanweisungen würden Sie nicht nur die alte Adresse überschreiben, sondern auch einen Teil der folgenden.

Warten Sie bitte noch, bevor Sie sich nun entsetzt von sequentiellen Dateien abwenden. Sequentielle Dateien können trotz all dieser Probleme sehr wohl vernünftig eingesetzt werden. Unter einer Voraussetzung: Gleich nach dem Start liest Ihr Programm die komplette Datei von Diskette und speichert die Adressen in einer Tabelle (Array). Die Adressen befinden sich nun zum Beispiel in folgender Form im Speicher (Basic):

```
N$(1) = "Maier"
V$(1) = "Georg"
S$(1) = "Im Tal 27"
P$(1) = "8000"
O$(1) = "München"

N$(2) = "Müller"
V$(1) = "Gerd"
S$(2) = "Aalweg 5"
P$(2) = "1000"
O$(2) = "Berlin"
...
```

Wenn sich alle Adressen im Speicher befinden, interessieren die Unzulänglichkeiten sequentieller Dateien nicht mehr. Auf der Suche nach Herrn »Müller« durchsuchen Sie nicht die Diskette, sondern das Stringarray N$(). Beim Ändern einer Adresse weisen Sie den entsprechenden Variablen neue Werte zu.

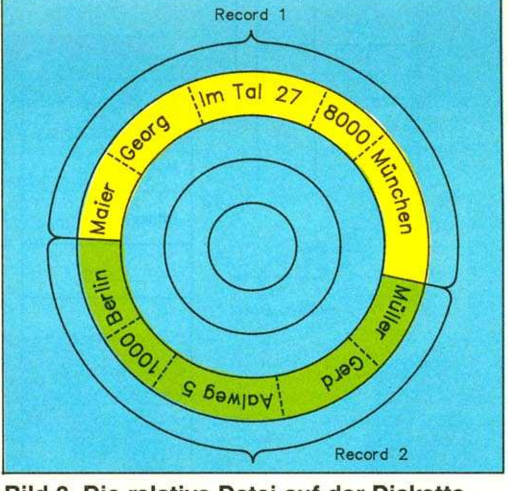


Bild 3. Die relative Datei auf der Diskette

Vor dem Beenden Ihres Programms sollten die — inzwischen möglicherweise veränderten — Adressen natürlich wieder auf der Diskette gespeichert werden. Kein Problem: Sie überschreiben einfach die alte Adreßdatei mit den neuen Adressen. Unter dieser Voraussetzung ist eine sequentielle Datei die einfachste Dateiform überhaupt. Am Programmanfang wird die komplette Datei eingelesen, am Programmende komplett gespeichert und ansonsten nur auf die gewohnte Weise im Computerspeicher mit Variablen gearbeitet. Das komplette Lesen und Speichern kann zwar bei umfangreichen Dateien recht lange dauern, aber bedenken Sie, daß beide Vorgänge nur einmal stattfinden.

Dafür können Sie zwischendurch stundenlang und ohne Zeitverluste mit Ihrer Datei arbeiten. Denn Zugriffe im Computerspeicher sind derart schnell, daß selbst Arrays mit 100 Adressen praktisch in »Nullzeit« bearbeitet werden.

Theoretische Abhandlungen sind nicht jedermanns Sache. Deshalb nun ein praktisches Beispiel. Vor dem Zugriff auf eine sequentielle Datei muß diese — wie auch ein altmodischer Karteikasten — geöffnet werden. Damit der Amiga weiß, welchen Dateikasten er öffnen soll, sind diese durch einen Dateinamen gekennzeichnet. In Basic öffnen Sie eine Datei mit der OPEN-Anweisung, in C mit FOPEN und in Pascal mit RESET oder REWRITE. Basic benötigt außer dem Dateinamen zusätzlich eine »Dateinummer« und eine Kennung, ob Daten in die Datei geschrieben oder daraus gelesen werden sollen (beides gleichzeitig ist mit sequentiellen Dateien nicht möglich).

```
REM Zum Schreiben öffnen
OPEN "TEST" FOR OUTPUT
AS #1
REM Zum Lesen öffnen
OPEN "TEST" FOR INPUT
AS #1
```

Für Speicherung und Lesevorgang nutzt Basic Varianten der PRINT- beziehungsweise INPUT-Anweisung.

```
REM *** Daten speichern
PRINT #1,"HALLO"
REM *** Daten lesen
LINE INPUT #1,A$
```

Die Daten befinden sich nach Ausführung der Lese-Anweisungen in der angegebenen Stringvariable (in diesem Fall A$). Nach dem letzten Schreib- oder Lesevorgang muß die Datei wieder geschlossen werden. Das geschieht mit

```
CLOSE #1
```

Beim Umgang mit sequentiellen Dateien werden also immer drei Schritte benötigt:

Von der Theorie zur Praxis

1. Datei öffnen
2. Beliebig viele Daten der Reihe nach speichern oder lesen
3. Datei schließen

Beim Lesen einer kompletten Datei allerdings taucht ein Problem auf: Woran erkennen wir, ob bereits das Dateiende erreicht ist? Zu diesem Zweck stellt jede Programmiersprache eine eigene Funktion bereit. In Basic ist es die EOF-Funktion.

```
v = EOF(dateinummer)
```

Diese Funktion übergibt den Wert Null, solange das Dateiende noch nicht erreicht, einen Wert ungleich Null, wenn es erreicht ist. Daher verwendet man zum Lesen einer Datei üblicherweise eine WHILE-Schleife nach folgendem Schema:

```
WHILE NOT EOF(dateinummer)
  Leseanweisung(en)
WEND
```

Das erste Basic-Demoprogramm (Listing 1) fordert Sie zunächst auf, zehn Namen einzugeben, die in N$(1) bis N$(10) abgelegt werden. Anschließend werden diese zehn Variablen in der Datei TEST gespeichert. Beachten Sie die Angabe der Dateinummer (eins) und die Richtung des Datenflusses (FOR OUTPUT = Speichern von Daten). Die Anweisung CLOSE schließt die Datei TEST wieder.

Zeile 260 löscht die Arrayvariablen, bevor die Datei nun zur Kontrolle komplett gelesen wird. Wieder benutzen wir die Dateinummer 1. Diesmal sollen Daten gelesen werden (FOR INPUT). In der WHILE-

Schleife liest die Anweisung LINE INPUT String für String ein, bis die EOF-Funktion einen Wert ungleich 0 übergibt.

Der Umgang mit sequentiellen Dateien ist sehr bequem, wenn man die vorgestellte Methode verwendet (die Datei komplett in den Speicher laden, dort bearbeiten, und vor dem Verlassen des Programms wieder komplett speichern). Allerdings kann die sequentielle Datei keinesfalls größer werden als die freie Speicherkapazität im Computer. Und wer hat schon die MBytes in seinem Computer?

Die Lösung dieses Problems lautet: Direktzugriffsdatei. Anders als bei der sequentiellen Methode wird beim direkten Zugriff nicht die gesamte Datei im Speicher gehalten, sondern immer nur der gerade bearbeitete Datensatz. Daher ist die Größe einer Direktzugriffsdatei nur durch die Kapazität des Massenspeichers beschränkt. Wenn Sie zum Beispiel eine 20-MByte-Festplatte besitzen, können tatsächlich 20 MByte an Daten im Direktzugriff vernünftig verwaltet werden.

Dateien im direkten Zugriff

Der Name sagt es bereits: Direktzugriffsdateien erlauben den direkten Zugriff auf jeden beliebigen Satz der Datei. Daher ist es im Gegensatz zu sequentiellen Dateien nicht nötig, die Datei komplett in den Hauptspeicher zu laden. Wird ein bestimmter Datensatz benötigt, kann direkt auf ihn zugegriffen werden, unabhängig davon, ob er sich am Anfang, in der Mitte oder am Ende der Datei befindet.

Direktzugriffsdateien sind intern in Blöcke konstanter Länge — die sogenannten »Records« — unterteilt. In jedem Record befindet sich genau ein Datensatz. Records besitzen fortlaufende Nummern (Record 1, Record 2,...). Der Zugriff auf einen bestimmten Datensatz erfolgt über die Nummer des Records, in dem er gespeichert ist (Bild 3).

Da alle Records die gleiche Länge besitzen, kann das Betriebssystem des Amiga die Position jedes beliebigen Records der Datei errechnen und den Schreib-/Lesekopf des Laufwerks direkt zu jenem Record bewegen (bei sequentiellen Dateien aufgrund der variablen Datensatzlänge nicht möglich).

Wenn Sie Bild 3 (Direktzugriffsdateien) mit Bild 2 (sequentielle Dateien) vergleichen, stellen Sie fest, daß Direktzugriffsdateien recht verschwenderisch mit der Diskettenkapazität umgehen. Die einzelnen Felder eines Datensatzes sind nicht komprimiert gespeichert, sondern zwischen den einzelnen Feldern befindet sich ein »Leerraum«. Ursache dieser verschwenderischen Speichertechnik ist die benötigte feste Recordlänge. Wie lang ein Record ist, bestimmen Sie indirekt durch die Festlegung der Feldlänge. Beispiel:
Feld Name: 10 Zeichen
Feld Vorname: 10 Zeichen
Feld Straße: 15 Zeichen
Feld Plz: 4 Zeichen
Feld Ort: 15 Zeichen

Schlüssel-wort
Satz-nummer
Bauer 003
Maier 001
Müller 002
sequentielle Datei
Maier Georg ... Record 1
Müller Gerd ... Record 2
Bauer Fritz ... Record 3
... Record 4
...
relative Datei

Bild 4. Zugriffsschema einer indexsequentiellen Datei

In diesem Beispiel besitzt jeder Record eine Länge von 54 Zeichen. Ist eine Adresse kürzer, wird das Feld mit Leerzeichen aufgefüllt. Daher sollten Sie »vernünftige« Feldlängen angeben und nicht etwa 100 Zeichen für den Inhalt des Feldes »Postleitzahl«. Die Diskette ist sonst sehr schnell voll.

Tragischer ist allerdings der umgekehrte Fall, die Angabe zu kleiner Feldlängen. Die Recordlänge einer Direktzugriffsdatei läßt sich nicht nachträglich ändern. Haben Sie beispielsweise für das Feld »Name« nur zehn Zeichen reserviert, ist es nicht möglich, schnell mal die Recordlänge zu ändern, um den eigentlich zu langen Namen »Hubermaier-Schmörgl« zu speichern.

Einfache Beispiele in Basic

Die Behandlung relativer Dateien ist prinzipiell immer gleich, unabhängig von der Programmiersprache. Die Datei wird geöffnet und dabei die Recordlänge angegeben. Datensätze lassen sich mit der Angabe der gewünschten Recordnummer lesen oder speichern. Am Schluß der Dateiarbeit muß diese wieder geschlossen werden. In Basic sieht dieser Ablauf folgendermaßen aus:

Geöffnet wird auch eine relative Datei mit OPEN. Allerdings ist die Angabe Datenübertragungsrichtung (Schreiben oder Lesen) nicht mehr nötig. Aus einer geöffneten relativen Datei lassen sich abwechselnd Datensätze lesen oder in diese welche speichern. Die für die Adreßberechnung notwendige Recordlänge ist Bestandteil der OPEN-Anweisung: »OPEN "R", #dateinummer, Dateiname, Recordlänge«. Beispiel:

```
OPEN "R",#1,"REL-
TEST",54)
```

Für die Verwaltung relativer Dateien verwendet das Betriebssystem einen »Datenpuffer«. Je nach Programmiersprache unterschiedliche Anweisungen unterteilen diese Puffer in Abschnitte, die den einzelnen Feldern entsprechen. Basic nutzt die Anweisung »FIELD #dateinummer, Länge AS Stringvariable [Länge AS Stringvariable]...« dazu.

Das heißt, in der FIELD-Anweisung ist die Länge jedes einzelnen Feldes und je eine dem Feld zugeordnete sogenannte »Puffervariable« angegeben (als Puffervariable ist immer eine Stringvariable anzugeben). Auf das Beispiel mit den fünf Feldern »Name«, »Vorname« ... bezogen:

```
FIELD#1,10 AS F1$,10 AS
F2$,15 AS F3$,4 AS F4$,
15 AS F5$
```

Dem ersten Feld ist somit die Puffervariable <F1$> zugeordnet, dem zweiten Feld <F2$> und so weiter. Mit einer weiteren Anweisung — in Basic ist das LSET — kann nun jedes Feld einer Adresse in die zugeordnete Puffervariable übertragen werden: »LSET Stringvariable = Zeichenkette«. Auf diese Weise wird der Datenpuffer mit einem zu speichernden Datensatz Feld für Feld gefüllt. Basic verwendet die Anweisung »PUT #dateinummer, recordnummer«, um den Inhalt des Datenpuffers gezielt in einem bestimmten Record zu speichern. Beispiel:

```
PUT #1,132
```

Umgekehrt kann mit »GET #dateinummer, recordnummer« der Inhalt eines bestimmten Records in die Puffervariablen gelesen werden.

```
GET#1,132
```

Dieser Ablauf ist in anderen Programmiersprachen sehr ähnlich. Zum Beispiel wird in Pascal mit SEEK auf einen bestimmten Record »positioniert« und anschließend mit READ der Record gelesen oder mit WRITE Daten in dem Record gespeichert.

Der Index im Speicher

Listing 2 ist ein Programmbeispiel für eine relativ verwaltete Datei. Wieder verlangt das Programm zehn Namen. Nach der Eingabe speichert es jeden Namen in die Direktzugriffsdatei REL-TEST. Und zwar wird der erste Name in Record Nummer 1 gespeichert, der zweite Name in Record Nummer 2 und so weiter.

Als Recordlänge haben wir 30 gewählt, ein für übliche Namen sicher ausreichender

Entdecken Sie jetzt mit Völkner die Welt der Elektronik!

Jeder Artikel nur **DM 5,-.** Gleich ankreuzen.

Bitte Ihre Artikel ankreuzen, die ganze Seite heraustrennen und heute noch an Völkner absenden.



Schnell-verbindungs- und Meß-strippen: 10 Strippen
in 5 Farben: weiß, schwarz, rot, grün, gelb, mit beidseitig isolierten Krokodil-klemmen. 331 ☐



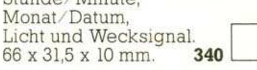
HiFi-Stereo-Kopf-hörer: Ohne Bügel,
federleicht, problemloser Sitz.
Enorme Klangfülle. 251 ☐



RENK-FORCE® Stecker-Netzgerät:
Universal-Netzgerät, passend in alle Steckdosen. 297 ☐



Mini-LCd-Digitaluhr mit Wecker
Stunde/Minute, Monat/Datum, Licht und Wecksignal. 66 x 31,5 x 10 mm. 340 ☐





Feinlötkolben
Leichte Ausführung für alle elektronischen Arbeiten. Schutzkontakt-Zuleitung, 220 V/25 W. 224 ☐



Elektronik-Lot 1 mm, die beliebte Menge, 15 m, in Kunststoff-Dose 260 ☐



Elektronischer Antennenverstärker für Autoantennen:
2stufiger Antennenverstärker für alle Pkw's mit 12-V-Bordnetz. 304 ☐





Sortiment Montage-schrauben
Enthält ca. 400 Teile übersichtlich in einem Sortierkasten. 322 ☐

5,-

Mikro-Pocket-Kamera 110:
Verblüffend klein, ...paßt in jede Hosentasche und ist völlig unkompliziert zu bedienen.
Ohne Film. 233 ☐



Elektro-nisches Taschen-Klavier
Ein Riesenspaß für groß und klein. Dieser Winzling spielt alle Melodien klar und rein. 279 ☐

5,-

Entöl-Pumpe:
Robuste Metall-ausführung mit hoher Absaugleistung. 359 ☐



Camera-Lamp:
Praktische Allzweck-lampe. Schwenkbares Oberteil. Batterie-/Netzbetrieb. 368 ☐



Als Dankeschön für Ihre Bestellung liegt dieses Universal-messer Ihrer Lieferung ohne gesonderte Berechnung bei sowie gratis der neue Völkner-Katalog.

☐ Herr ☐ Frau ☐ Frl.

Vorname, Name

Straße

PLZ/Ort

Die ganze Seite gleich ausschneiden und absenden an:

Völkner electronic GmbH+Co. KG
Postfach 5320
3300 Braunschweig

Sie erhalten die Sendung portofrei gegen Rechnung

Brandneue Bücher für Ihre Amiga-Bibliothek



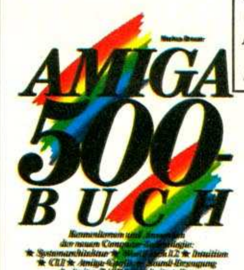
F. Kremser/J. Koch
Amiga-Programmierhandbuch
Eine Super-Einführung in die »Interna« des Amiga: Die wichtigsten Systembibliotheken werden ausführlich anhand von Beispielen in C erklärt.
1987, 387 Seiten, inkl. Diskette
Bestell-Nr. 90491, ISBN 3-89090-491-2
DM 69,–/sFr 63,50/öS 538,20

Dr. E. Huckert/F. Kremser
Amiga: C in Beispielen
Grundlagen, Programmsammlung, Aufruf der Systembibliotheken.
1987, 237 Seiten, inkl. Diskette
Bestell-Nr. 90539
ISBN 3-89090-539-0
DM 69,–/sFr 63,50/öS 538,20

NEU

A. Plenge
Amiga-3-D-Grafik und Animation
Eine leichtverständliche Anleitung für die Erstellung von dreidimensionalen Grafiken: Clipping, Perspektivische Projektion, Raytracing, Versteckte Linien, Schatten, Reflexion, 3-D-Editor.
1988, ca. 350 Seiten, inkl. Diskette
Bestell-Nr. 90526, ISBN 3-89090-526-9
DM 69,–/sFr 63,50/öS 538,20

M. Breuer, **Das Amiga-500-Buch**
Eine ausführliche Einführung in die Bedienung des Amiga 500. Mit einer ausführlichen Beschreibung des Amiga und seines Zubehörs. Das Handbuch dient als Nachschlagewerk beim alltäglichen Einsatz. Eine Pflichtlektüre für jeden, der sich für diesen Supercomputer interessiert.
1987, 489 Seiten, Bestell-Nr. 90522, ISBN 3-89090-522-6
DM 49,–/sFr 45,10/öS 382,20



M. Breuer, **Amiga-2000-Buch**
Eine Pflichtlektüre für jeden, der diesen Supercomputer erfolgreich einsetzen möchte: Systemarchitektur (OSA), Umgang mit der Workbench 1.2, CLI-Befehle, MS-DOS auf dem Amiga, Umgang mit der PC-Karte, Amiga-Grafik, Überblick über Amiga-Basic. Mit vielen Bildschirmfotos und Übersichtstabellen.
1987, 570 Seiten, Bestell-Nr. 90574, ISBN 3-89090-574-9
DM 59,–/sFr 54,30/öS 460,20

P. Wollschlaeger, **Amiga-Assembler-Buch**
Dieses Buch beweist, daß Assembler-Programmierung ganz einfach ist: Ein 68000er-Kurs mit vielen Beispielen. Mit ausführlichem Verzeichnis aller Systemroutinen, Anleitung für das Einbinden von Assembler-Routinen in Amiga-Basic und vielen Informationen über die Internas des Amiga-Betriebssystems. Mit Beispieldiskette.
1987, 329 Seiten, inkl. Diskette
Bestell-Nr. 90525, ISBN 3-89090-525-0
DM 59,–/sFr 54,30/öS 460,20



Bantam Books
Das Amiga-DOS-Handbuch für Amiga 500, 1000 und 2000
Die Pflichtlektüre für jeden Commodore-Amiga-Anwender und Programmierer: die deutsche Ausgabe der offiziellen Entwickler-Dokumentation zum Amiga-DOS-Betriebssystem, Version 1.2. Programmierung, interne Datenstruktur und Diskettenhandling. Alle Möglichkeiten des Systems, bis hin zum Multitasking, werden ausführlich behandelt.
1987, 342 Seiten, Bestell-Nr. 90465, ISBN 3-89090-465-3
DM 59,–/sFr 54,30/öS 460,20

Markt&Technik-Produkte erhalten Sie bei Ihrem Buchhändler, in Computerfachgeschäften oder in den Fachabteilungen der Warenhäuser.

Irrtümer und Änderungen vorbehalten.



Zeitschriften · Bücher
Software · Schulung

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0.
SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656,
ÖSTERREICH: Markt&Technik Verlag Ges.m.b.H, Große Neugasse 28, 1010 Wien, Telefon (0222) 5879455,
Rudolf Lechner&Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526, Ueberreuter Media Handels- und Verlagsges. mbH (Großhandel), Laudongasse 29, A-1082 Wien, Telefon (0222) 481543-0

Fragen Sie bei Ihrem Buchhändler nach unserem kostenlosen Gesamtverzeichnis mit über 500 aktuellen Computerbüchern und Software. Oder fordern Sie es direkt beim Verlag an!

802124

Wert. Nach dem Öffnen wird der Datenpuffer mit FIELD unterteilt. Bei nur einem Feld gibt es allerdings nicht viel zu unterteilen: Eine einzige Puffervariable »F1$« mit einer Länge von 30 Zeichen reicht aus. In einer Schleife erfolgt die Eingabe der Namen und deren Speicherung in »N$«. LSET überträgt den Inhalt von »N$« jeweils in die Puffervariable und PUT schreibt den Pufferinhalt anschließend in die Records 1, 2, 3 und so weiter. In umgekehrter Reihenfolge liest das Programm die Inhalte der Records 10 bis 1 dann wieder in den Datenpuffer (»F1$«) ein und gibt sie auf dem Bildschirm aus. Zum Abschluß wird die Datei wie üblich geschlossen. Aufgrund des direkten Zugriffs und der festen Recordlänge ist es übrigens nicht nötig, relative Dateien komplett in den Speicher zu laden und dort zu bearbeiten.

```
100 CLS : PRINT "Geben Sie bitte 10 Namen ein."
110 FOR i=1 TO 10
120    INPUT "Name";N$(i)
130 NEXT i
140 :
200 OPEN "test" FOR OUTPUT AS #1
210 FOR i=1 TO 10
220    PRINT#1,N$(i)
230 NEXT i
240 CLOSE #1
250 :
260 FOR i=1 TO 10: N$(i) = "": NEXT i
270 :
300 OPEN "test" FOR INPUT AS #1
310 WHILE NOT EOF(1)
320    LINE INPUT#1,N$(AD + 1): PRINT N$(AD + 1)
330    AD = AD + 1
340 WEND
350 CLOSE #1
```

Listing 1. Programm bei sequentieller Dateiorganisation

```
100 CLS : PRINT "Geben Sie bitte 10 Namen ein."
110 OPEN "r", #1, "test", 30
120 FIELD #1, 30 AS F1$
130 FOR i=1 TO 10
140    INPUT "Name";N$
150    LSET F1$ = N$
160    PUT#1,i
170 NEXT i
180 :
200 FOR i=10 TO 1 STEP-1
210    GET#1, i
220    PRINT F1$
230 NEXT i
240 CLOSE#1
```

Listing 2. Programm bei relativer Dateiorganisation

Der Schlüssel zur Datei

Einen Nachteil der Direktzugriffsdateien haben wir bisher verschwiegen. Angenommen, Sie suchen wieder mal Herrn Müller. Wenn Sie die zugehörige Recordnummer kennen, können Sie seine Adresse direkt einlesen. Leider wissen Sie normalerweise nicht, in welchem Record sich diese Adresse befindet.

Wir haben behauptet, es sei unnötig, Direktzugriffsdateien komplett in den Speicher zu laden. Wenn sich die Adreßdatei ausschließlich auf der Diskette befindet, bleibt Ihnen nichts anderes übrig, als die Direktzugriffsdatei »sequentiell«, also Record für Record, nach Herrn Müller abzusuchen.

Ist die Datei nicht gerade klein, kann dieser Vorgang sehr lange dauern. Es stellt sich also die Frage: Was haben wir von der Möglichkeit, direkt unter Angabe der Recordnummer zuzugreifen, wenn wir nicht wissen, in welchem Record sich eine bestimmte Adresse befindet? Die Lösung des Problems ist die »index-sequentielle« Organisation. Eine index-sequentielle Datei ist im Grunde weder eine neue Dateiform noch eine neue Zugriffsart, sondern eine geschickte Kombination der sequentiellen und relativen Verfahren.

In der Praxis ist eine Komponente der gesuchten Daten meist bekannt. Bei einer (Adressen-)Dateiverwaltung ist diese Komponente üblicherweise das Feld »Name«. Zum Beispiel suchen Sie die Telefonnummer oder die Anschrift von Herrn »Müller«. In beiden Fällen ist das Feld »Name« ein »Schlüsselfeld«. Es enthält einen »Schlüssel«, eben den Namen »Müller«, der Ihnen Zugang zu den restlichen Teilen der Daten verschaffen kann.

Bei index-sequentiellen Dateien befindet sich die eigentliche Datei in einer Direktzugriffsdatei auf der Diskette. Zusätzlich legt man aber von jedem Datensatz den Inhalt des Schlüsselfeldes und die Nummer des Records im Hauptspeicher ab. Nehmen wir das Beispiel einer Adreßverwaltung (Bild 4). Im Hauptspeicher wird für jeden Datensatz der Inhalt des Schlüsselfeldes »Name« und die Recordnummer der zugehörigen Adresse gespeichert (zum Beispiel in zwei Tabellen, einem String- und einem Integerarray):

Name	Record
Maier	1
Müller	2
Bauer	3

Bei jedem Suchvorgang wird nun »indirekt«, über das Schlüsselfeld, auf die Direktzugriffsdatei zugegriffen. Angenommen, Sie suchen die Telefonnummer von Herrn »Müller«. Ihr Programm fragt Sie nach dem Namen. Sie geben ein: »Müller«. Das Programm durchsucht sequentiell das Array, in dem die Namen gespeichert sind. Das Programm findet »Müller« als zweiten Eintrag. Der zugehörige zweite Eintrag im Array mit den Recordnummern enthält eine Zwei. Also weiß Ihr Programm, daß sich die Adresse von Herrn »Müller« in Record Nummer 2 befindet. Und auf diesen Record greift es nun direkt zu.

Der Vorgang verläuft analog, wenn sich zum Beispiel die Telefonnummer von Herrn »Bauer« ändert. Das Programm fragt nach dem Namen der zu ändernden Adresse. Sie geben »Bauer« ein und das Programm findet im Speicher nach einem blitzschnellen Durchsuchen des Arrays »Bauer« als dritten Eintrag und 3 als zugehörige Recordnummer. Wenn Sie nun die geänderte Adresse mit der neuen Telefonnummer eingeben, wird Ihr Programm den Inhalt von Record Nummer 3 überschreiben.

Der zusätzliche Aufwand zur index-sequentiellen Verwaltung hält sich in Grenzen. Bei jedem Neueintrag speichern Sie wie gewohnt die Adresse im ersten noch nicht belegten Record der Direktzugriffsdatei. Zusätzlich speichern Sie den Inhalt des Schlüsselfeldes und die Nummer des Records in den ersten freien Elementen Ihrer beiden Arrays. Vor dem Beenden des Programms sichern Sie die Inhalte der beiden Arrays in einer sequentiellen Datei. Wenn Sie das Programm am nächsten Tag erneut benutzen, laden Sie zuerst diese sequentielle Datei wieder in den Speicher.

Index-sequentielle Dateien bieten auch beim Sortieren enorme Vorteile. Nehmen wir an, Sie wollen die Adressen nach den enthaltenen Namen sortieren. Sortieren beruht auf mehrfachem Vertauschen von Elementen.

Beim indexsequentiellen Verfahren vertauschen Sie Elemente der beiden Arrays im Speicher. Sie sortieren einfach das Array, das die Namen der Adressen enthält. Und tauschen bei jedem Vertauschen zweier Namen auch die zugehörigen Recordnummern, die in einem Integer-Array gespeichert sind, aus. Angewandt auf die im vorigen Beispiel verwendeten Adressen ergibt sich nach dem Sortieren im Hauptspeicher folgendes Bild (Indizes und zugehörige Recordnummern):

Name	Record
Bauer	3
Maier	1
Müller	2

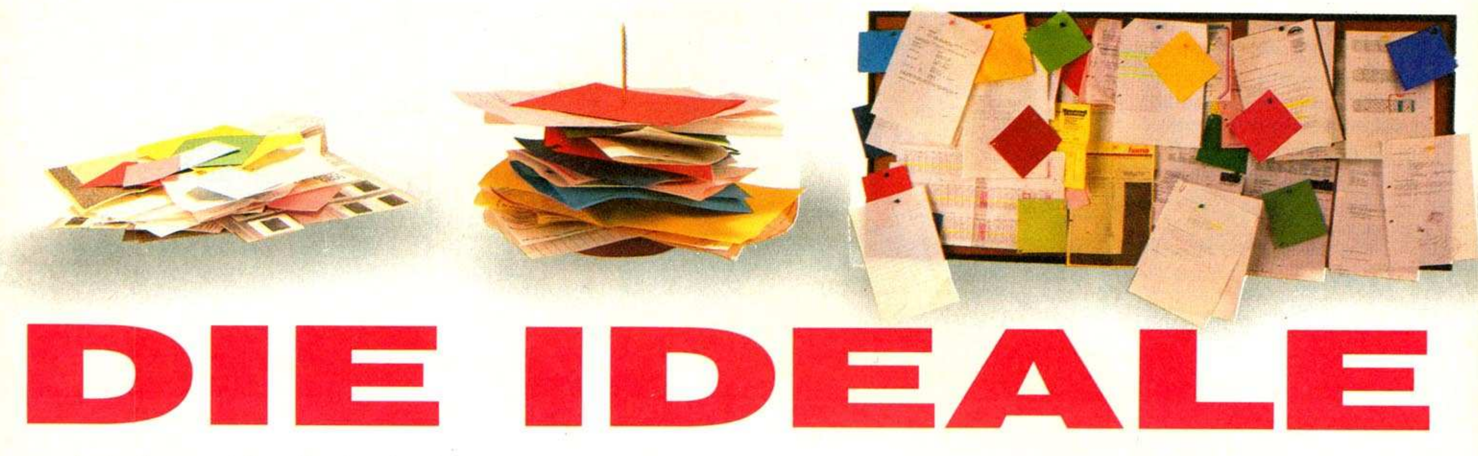
Sie haben sicher erkannt, daß die Direktzugriffsdatei auf der Diskette nicht verändert werden braucht. Nur im Speicher kommt das bisher letzte Arrayelement »Bauer« alphabetisch richtig an den Anfang des Arrays und die zugehörige Recordnummer 3 ebenfalls an den Anfang des Integerarrays. Obwohl die Direktzugriffsdatei immer noch unsortiert ist, läßt sich problemlos eine alphabetisch nach Namen sortierte Liste der Adressen ausgeben. Sie greifen der Reihe nach auf die sortierten Elemente Ihrer Arrays zu und lesen jeweils den Record ein, dessen Nummer im Integer-Array gespeichert ist. Also geben Sie zuerst die Adresse in Record 3, dann die in Record 1 und anschließend die im Record Nummer 2 gespeicherte Adresse aus.

Damit wären wir am Ende unserers Exkurses in die Dateiverwaltung. Vielleicht wollten Sie ja schon lange mal eine Liste Ihrer Platten, Kassetten oder Zeitschriften-Artikel anlegen. Der Amiga kann Ihnen langwierige Routinetätigkeiten dabei abnehmen.

(Said Baloui/pa)

Welche Forderungen können Sie an eine moderne Dateiverwaltung oder Dankenbank auf dem Amiga stellen?

Um diese Frage zu beantworten, greifen wir zu einem Kunstgriff. Wir werden Ihnen im folgenden eine Datenbank vorstellen, die es (noch) nicht gibt. Dieses Programm erfüllt die Anforderungen, die man heute an eine Datenbank auf dem Amiga stellen kann. Wir sind dabei nicht so vermessen, uns den Amiga in einem mittelständischen Unternehmen netzwerkfähig im Verbund mit Anlagen der mittleren Datentechnik vorzustellen. Das ist sicherlich kurzfristig nicht zu realisieren. Unsere ideale Datenbank ist auf die Anwender zugeschnitten, die jetzt mit dem Amiga arbeiten oder — Industriestandard hin oder her — kurz vor der Anschaffung dieses Computers stehen. An den Leistungsmerkmalen von »DB 2000«, so nennen wir diese fiktive Datenbank, messen wir anschließend die realen Datenmanager. Superbase, Datamat, GoAmiga Datei und Organize. DB 2000 wird damit unser Referenzprogramm für Datenmanager. Der Vergleich hilft Ihnen, sich für das richtige Programm zu entscheiden. Bei der Entwicklung von DB 2000 haben wir natürlich auf besonders gelungene Elemente unter anderem der vorgestellten »Realos« zurückgegriffen.



DIE IDEALE

Bevor wir mit unserer Vorstellung beginnen, sollte allerdings die Bedeutung der Fachausdrücke Datenbank und Dateiverwaltung geklärt werden. Zu Beginn der Mikrocomputer-Ära entstanden auf den »kleinen Rechnern« sehr schnell Programme, mit denen sich Adressen verwalten ließen. Diese Adreßverwaltungen besaßen eine festdefinierte Eingabemaske, innerhalb derer die Teile einer Adresse eingegeben werden konnten. Bei den Dateiverwaltungen fällt diese Beschränkung auf eine bestimmte Art Daten weg. Hier lassen sich Eingabemaske und damit Anzahl und Art der zu verwaltenden Daten frei definieren. Gemeinsam ist beiden Programmen allerdings, daß sie nur eine Datei gleichzeitig bearbeiten können. Eine Datenbank wiederum enthält in mehreren Dateien jeweils verschiedenartige Informationen. Trotz der Verschiedenheit haben die Daten eine Beziehung (Relation) zueinander (Beispiel: Kundendatei, Artikeldatei, Verkaufsdatei). Durch Eingabe von einem oder mehreren, eventuell auch miteinander logisch verknüpften Stichworten lassen sich komplexe Auswertungen herstellen.

Die Module von DB 2000

Wir werden im folgenden noch erfahren, wie das funktioniert. In unserem Grundlagenartikel auf Seite 18 finden Sie übrigens weitere Informationen zu diesem Thema.

DB 2000 setzt sich in der Regel aus vier Komponenten zusammen: Datendatei, Dateidefinition, Index- und Maskendateien. Die Datendatei enthält die vom Anwender eingegebenen Daten. In der Indexdatei befinden sich Informationen darüber, an welcher Stelle der Datei sich bestimmte Daten (zum Beispiel die Adresse von Herrn Blüm oder Frau Carlsson) befinden. Darüber hinaus benötigt eine Datenbank Angaben über Art und Anzahl der Informationen, die gespeichert werden sollen. Diese befinden sich in der Dateidefinition. Die Maskendateien schließlich enthalten Bildschirm- und Druckermasken, also die Ein- und Ausgabeformulare für die Daten. DB 2000 legt nur auf Wunsch Index- und Maskendateien an.

Vor der ersten Arbeit mit DB 2000 steht also die Definition einer Datei. Nach Anwahl der entsprechenden Menüfunktion öffnet sich ein Fenster für die Felddefinition. Für den Namen des Feldes steht ein Eingabefeld zur Verfügung. Die Betätigung von <Return> nach der Eingabe genügt und schon ist ein Textfeld mit einer Länge von 20 Zeichen definiert. Das Programm löscht das Eingabefeld und weitere Felder können eingegeben werden. Bereits definierte erscheinen in einer Tabelle.

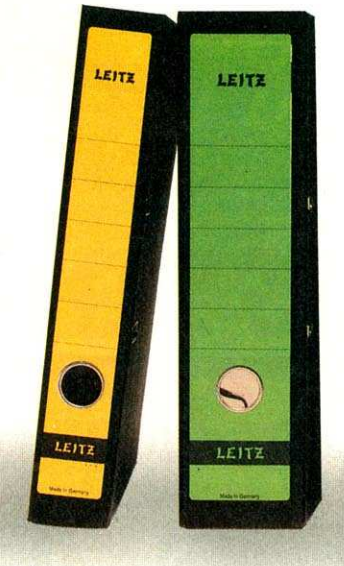
DB 2000 kennt die Datentypen Text, Numerisch, Datum, Zeit und Extern und Telefon. Felder vom Typ Extern enthalten die Namen von Grafik- oder Sound-IFF-Dateien. Wir erläutern an anderer Stelle noch den einfachen Mechanismus, der diese Daten zur Anzeige beziehungsweise zum Ablauf bringt. Neben dem Primärattribut, also dem Datentyp, lassen sich weitere Sekundärattribute vergeben. Dies sind im einzelnen: Eingabe, Errechnung, Überprüfung, Erwartet, Repetier und Automatik. Bei der Definition sind alle Felder zunächst Eingabefelder. Mit dem Attribut »Errechenbar« versehene Felder sind in der Regel nicht Teil der Dateidefinition. Sie werden in den Masken definiert und bei der Ausgabe benutzt, um Summenberechnungen oder andere Auswertungen zu plazieren. DB 2000 wertet sowohl aus Referenzen auf andere numerische Felder, Konstanten, den fünf mathematischen Operatoren sowie Funktionen bestehende numerische Klammerausdrücke als auch Zeichenkettenausdrücke aus. Beispiel: »ROUND(Summe*(1-Rabatt/100),2)« rundet eine um einen Prozentsatz verminderte Zahl auf zwei Stellen hinter dem Komma auf. Solche Felder werden auch als abgeleitete Felder bezeichnet, da sich ihr Wert von anderen Feldern ableitet.

»Überprüfung« veranlaßt DB 2000, in diesem Feld gemachte Eintragungen mit vom Anwender erlaubten zu vergleichen. So können beispielsweise die Eingaben für das Feld »Anrede« auf die üblichen Angaben beschränkt werden. Der Inhalt der Felder läßt sich unter Anwendung aller von einer Programmiersprache bekannten Vergleichsoperatoren mit dem Wert eines Ausdrucks vergleichen. Das Attribut »Erwartet« erzwingt bei der späteren Dateneingabe einen Eintrag in dieses Feld. In mit dem Attribut »Repetier« belegte Felder überträgt DB 2000 bei einer leeren Dateneingabe (nur <Return> drücken) automatisch die beim vorherigen Datensatz in dieses Feld eingegebenen Daten. Das Attribut »Automatik« ist nur für Felder vom Typ Datum und Zeit zugelassen. Sie enthalten »automatisch« das aktuelle Datum und die aktuelle Zeit. Voraussetzung dafür ist natürlich, daß die Systemzeit korrekt eingestellt wurde. Felder vom Typ »Telefon« schließlich stellen sicher, daß für alle Telefonnummern dasselbe Format benutzt wird.

Da mit dem oben beschriebenen Mechanismus bei der Felddefinition automatisch Textfelder entstehen, muß bei Wahl eines anderen Datentyps das bereits definierte Feld in der Tabelle angeklickt werden. Es ist dann selektiert und der Datentyp kann durch Anklicken der entsprechenden Gadgets (Symbols) in den gewünschten Datentyp geändert werden. Die Entwickler von DB 2000 haben an dieser Stelle keine weiteren Formatdefinitio-

Wie sieht die ideale Datenbank aus? Wir stellen Ihnen DB 2000 vor. Unsere ideale Datenbank. An dieser fiktiven Datenbank müssen sich Superbase, Datamat, GoAmiga! Datei und Organize messen lassen.







DATENBANK

nen (für Form des Datums, numerischer Werte...) vorgesehen. Alle Felder besitzen Standardlänge und -Format. Der Anwender kann diese Formate bei der Maskendefinition nachträglich verändern.

Nach der Eingabe der Felddefinition legt DB 2000 automatisch die Definitionsdatei und eine leere Datendatei an. Auf dem Bildschirm bildet das Programm die Standardeingabemaske ab — die Dateneingabe kann beginnen. Was uns das Programm allerdings als Eingabeformular anbietet, wird dem kreativen Computerfan (und das sind ja fast alle Besitzer des Amiga) kaum genügen.

Ein komfortabler Maskeneditor

Schwarz auf Weiß bildet das Programm den Dateinamen als Titel ab und plaziert die einzelnen Feldnamen vor die blau unterlegten und gleich langen Eingabefelder. Mit der Gestaltung der Eingabemaske kommen wir zur ersten Finesse von DB 2000. Mit Anwahl des Menüpunktes »Datei, Neu, Maske« öffnet das Programm ein bildfüllendes Fenster. Es enthält die eben erwähnte Standardmaske und eine Schalttafel (Gadget) mit verschiedenen Funktionsymbolen (ähnlich wie in Bild 1).

Durch Anklicken eines Feldes wird dieses selektiert. Selektierte Felder lassen sich verschieben und vergrößern. Ein einschaltbares Raster erleichtert die Positionierung der Felder durch Variierung der möglichen Positionen in 8-Punkt-Schritten. Bei der Veränderung der Feldgröße blendet DB 2000 eine Anzeige ein, welche laufend die vom Feld beanspruchte Anzahl Spalten und Zeilen (!) sowie die dadurch bedingte Zeichenkapazität anzeigt. Das Anlegen eines Feldes über mehrere Zeilen ergibt nur bei Textfeldern einen Sinn. Gerade für solche Felder hat das Programm noch einige Überraschungen für Sie parat. Doch dazu später.

Ein kurzes Anklicken eines selektierten Feldes bringt eine weitere Schalttafel auf den Bildschirm. Hier kann der Anwender jetzt das Format der Felder definieren oder aber die Sekundärattribute ändern. Prinzipiell bietet DB 2000 mehrere Formattypen an. Der Anwender kann diese mit einem Mausklick aus einer Tabelle heraus aktivieren und gegebenenfalls modifizieren. Für die Veränderung stehen Kennbuchstaben zur Verfügung. Beispiel für numerische Werte: »*-%4,2« stellt ein Zahlenformat ein, bei dem 1. nicht ausgenutzte Stellen vor dem Komma mit Sternchen ausgefüllt, 2. nur im Falle negativer Werte ein Vorzeichen angezeigt und 3. die Zahlen mit vier Vorkomma- und zwei Nachkommastellen durch ein Komma getrennt ausgegeben werden (%0,0 -> höchste Genauigkeit). Weitere Kennziffern ersetzen den Wert Null gegen eine Leerstelle, drucken auch positive Vorzeichen oder geben den Wert in wissenschaftlicher Notation (Beispiel 10E2) aus. Der Formatkette vorangestellte oder folgende Zeichenketten gibt das Programm unverändert aus. So lassen sich x-beliebige Währungssymbole positionieren. Ähnlichen Komfort bietet die Bestimmung des Datum- und Zeitformats. Beim Datum beispielsweise bestimmen Kennziffern die Form der Monatsangabe (Zahl oder Name), des Jahres (1988,88,'88), ob der Tagesname (So, Sonntag, Sonntag,den) vor das Datum gesetzt wird oder nur eine Zahl (070288) das Datum repräsentieren soll. Durch vorgestellte Zeichenketten lassen sich Formate wie »München, den 7.2. '88« problemlos realisieren. Textfelder schließlich können rechts- oder linksbündig beziehungsweise zentriert formatiert werden.

Grafikelemente des Editors

Für die weitere Gestaltung der Eingabemaske stellt DB 2000 eine Reihe vierfarbiger grafischer Elemente zur Verfügung. So lassen sich eng zusammengehörige Datenfelder durch das Ziehen von Linien oder Setzen von Rechtecken (auch gefüllt) mit vier unterschiedlichen Strichstärken von anderen separieren. Das Programm plaziert mit den Zeichensätzen des Amiga und den von Textverarbeitungen gewohnten Attributen (fett,kursiv,breit,unterstrichen) gestaltete Texte an beliebige Stellen der Masken. Der Editor arbeitet objektorientiert. Anders als bei Malprogrammen, wie zum Beispiel DPaint, können daher »Objekte« (einschließlich Felder) durch Selektieren und Anklicken des Mülleimers ohne »Beschädigung« des Umfeldes entfernt werden.

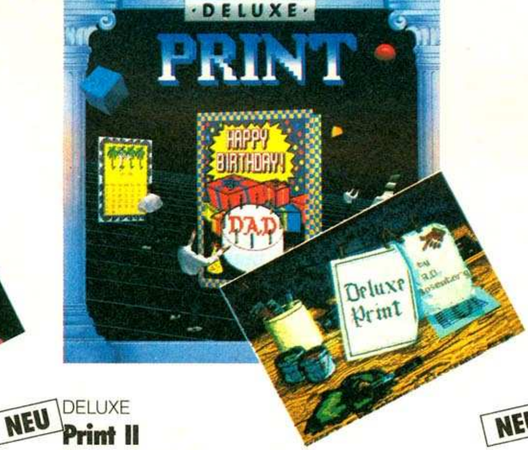


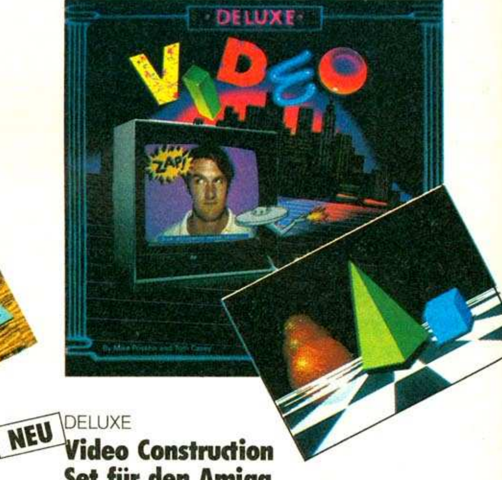
ELECTRONIC ARTS™

ELECTRONIC ARTS DELUXE-Serie

Exklusivvertrieb bei Markt&Technik







NEU

DELUXE
Paint II/deutsch für den Amiga

Deluxe Paint II ist eines der außergewöhnlichsten Grafikprogramme auf dem Softwaremarkt. Im Vergleich zu seinem Vorgänger ist Deluxe Paint II in der Lage, in der Textfunktion Fett- und Kursivschrift sowie Unterstreichungen einzubauen, perfekte Manipulation in der dritten Dimension darzustellen oder wichtige Parameter wie Längsdruck, Schwarzweißgraustufen- oder Farbdruck, Ränder und vieles mehr unmittelbar vor dem Druck per Mausklick zu verändern.
Hardware-Anforderungen:
Amiga (512 Kbyte) und Farbmonitor.
Bestell-Nr. 52584
DM 249,–* (sFr 225,–*/öS 2490,–*)
Gegen Einsendung Ihrer englischen Originaldiskette und gegen Vorauskasse erhalten Sie ein Update auf die deutsche PAL-Version von Paint II.
Bestell-Nr. 52584U
DM 135,–* (sFr 122,–*/öS 1350,–*)

Die ideale Ergänzung zu Deluxe Paint II:
Fertige Bilder und vorgefertigte Muster, die Sie frei nach Ihren Wünschen zusammenstellen können:

DELUXE
Seasons & Holidays
Bestell-Nr. 52580
DM 29,–* (sFr 24,–*/öS 290,–*)

DELUXE
Art Paint II
Bestell-Nr. 52581
DM 29,–* (sFr 24,–*/öS 290,–*)

NEU

DELUXE
Print II für den Amiga

und ein grafikfähiger Drucker verwandeln den Amiga in eine Druckmaschine. Sie können Karten, Poster, Briefköpfe und vieles mehr auf einfachste Weise entwerfen und ausdrucken. Besitzer eines Farbdruckers können ihr Werk auch in Farbe aufs Papier bringen. »Deluxe Print« ist kompatibel zu »Deluxe Paint«. Das bedeutet, daß man Grafiken zwischen den Programmen austauschen kann.
Hardware-Anforderungen:
Amiga (512 Kbyte) und Farbmonitor.
Bestell-Nr. 52566
DM 99,–* (sFr 89,–*/öS 990,–*)

DELUXE
Music/deutsch für den Amiga
Das professionelle Musikprogramm für Ihren Amiga!
Hardware-Anforderung: Amiga (512 Kbyte).
Bestell-Nr. 52579
DM 199,–* (sFr 179,–*/öS 199,–*)

NEU

DELUXE
Video Construction Set für den Amiga

Dient zum einfachen Entwerfen und Zusammenstellen von animierten Grafik-Sequenzen. Sie können so Videofilme mit Computergrafik versehen und regelrechte Computer-Videoclips zusammenstellen.
Das Programm ist ebenfalls kompatibel zu »DELUXE PAINT« und »DELUXE PRINT«.
Hardware-Anforderungen:
Amiga (512 Kbyte) und Farbmonitor.
Bestell-Nr. 52567
DM 199,–* (sFr 179,–*/öS 1990,–*)
Lieferbar: 1. Quartal 1988

Das Buch zur Software
Markus Breuer
DELUXE Grafik mit dem Amiga
1986, 370 Seiten
ISBN 3-89090-412-2
Best.-Nr. 90412
DM 49,– (sFr 45,10/öS 382,20)
Erhältlich bei Ihrem Buchhändler

* Unverbindliche Preisempfehlung



Zeitschriften · Bücher
Software · Schulung

Deluxe Paint, Deluxe Print und Deluxe Video erhalten Sie in den Fachabteilungen der Kaufhäuser, in Computershops, bei Commodore-Fachhändlern oder direkt beim Verlag gegen Vorauskasse.

803228

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0
Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656. ÖSTERREICH: Markt&Technik Verlag Ges.m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 5879455; Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526.

Damit ist die erste kreative Phase bei der Dateiarbeit beendet. Nun kommt erst mal eine Weile reine Routine — die Eingabe der Daten: Eingabemodus einschalten, Datensatz eingeben und mit <Amiga s> den Satz speichern. Das ist sicherlich der zeitaufwendigste Teil bei der Anlage einer Datenbank. Durch den geschickten Einsatz der oben erwähnten Repetierfelder können Sie sich diese Tätigkeit erleichtern. An dieser Stelle kommt nun das Anlegen von Feldern über mehrere Zeilen zum Tragen. Beispiel: Sie haben eine Datei mit sechs Feldern für Name und Adresse plus ein Feld — nennen wir es »Text« — plus ein Automatikfeld vom Typ Datum angelegt. Das Feld »Text« ist 30 Zeilen mal 80 Spalten groß. Nun ordnen Sie die Felder in der Ausgabemaske an wie in einem Brief, schreiben in das Feld »Text« den Brieftext, speichern Ihr Dokument und drucken es anschließend oder später aus. DB 2000 unterstützt Wordwrap (automatischer Umbruch von Wörtern, die nicht mehr komplett in eine Zeile passen) und besitzt wie Textverarbeitungsprogramme auch Funktionen zum Ausschneiden, Kopieren und Einfügen von vorher markierten Textteilen.

Dateneingabe bei DB 2000

Die freie Belegbarkeit der Funktionstasten mit oft verwendeten Kurztexten (Phrasen) kann hier sinnvoll genutzt werden. Wollten Sie nicht schon immer eine einfache Textverarbeitung für Ihre »paar« Briefe haben?

Ihre Adreß-, Medienverwaltung oder was auch immer hat nun langsam Gestalt angenommen. Selbst nach der Eingabe nur weniger Datensätze läßt sich mit den Daten experimentieren. Unterhalb des Arbeitsbildschirmes befindet sich hierfür eine Steuerleiste mit Symbolen, die an die Steuerfunktionen eines Kassettenrecorders erinnern. In der Tat haben sie auch gleichartige Funktionen. Da sind zum Beispiel die Tasten schneller Vorlauf/Rücklauf. Damit können Sie durch den Datenbestand blättern. Die Pausentaste läßt Sie bei einem bestimmten Datensatz verweilen und »Stop« beendet schließlich den Ablauf. Anklicken zweier weiterer Schalter veranlaßt DB 2000, den nächsten oder vorherigen Satz anzuzeigen. Das ist zwar alles ganz nett, aber dafür haben wir die Datenbank nicht angelegt.

Wie hieß denn noch gleich der Schauspieler in dem Spionagefilm mit dem Flugzeug ... Rußland ...? Wir löschen mit einem Schalter den aktuellen Datensatz unserer Medienverwaltung und geben in das Feld Inhalt ein: LIKE "*Flugzeug*" UND LIKE "*Rußland*", geben die Kombination <Amiga s> ein, hoffen, daß sich in unserer Datenbank nicht viele Filme befinden, die sich mit dieser Thematik befassen und erhalten wenige Sekunden später eine Beschreibung des Filmes »Firefox« mit Inhalt in Stichworten, Hauptdarsteller und Regisseur. Natürlich können wir auch mit »= "Meier"« die Adresse eines Schulkameraden ermitteln.

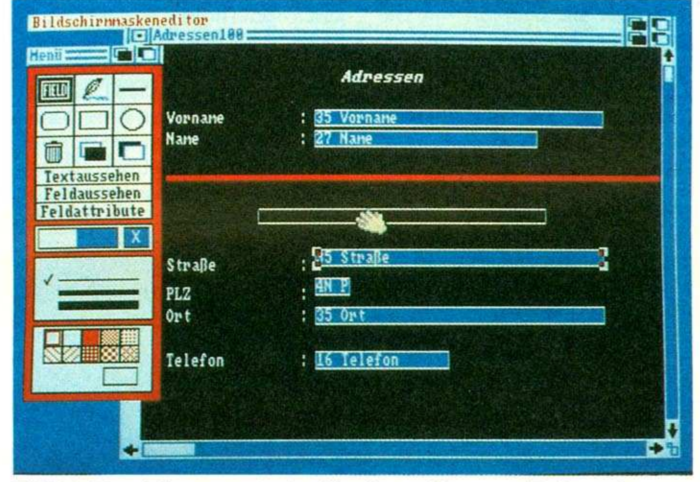


Bild 1. Der leistungsstarke Maskeneditor von Datamat

Das Schlüsselwort LIKE (so ähnlich) bedeutet, daß DB 2000 nicht auf Groß- oder Kleinschreibung achten soll. Die Sternchen vor und hinter den Suchkriterien stehen für beliebigen Text, der im Feld »Inhalt« vor und hinter den Worten »Flugzeug« und »Rußland« stehen kann. Hätten wir statt »UND« das Wort »ODER« verwendet, so hätte das Programm bereits Filme gemeldet, die sich mit der Steinzeit Rußlands oder der Kanalüberquerung von Lindbergh befassen. Anders die Folge »= Meier«. Sie fordert absolute Übereinstimmung und so findet DB 2000 auch nur einen »Meier«. Mit der Suchfunktion »global« lassen sich Zeichenfolgen ohne Festlegung auf ein Feld im gesamten Satz oder der ganzen Datei suchen.

Nachdem wir nun den passenden Film für unseren Videoabend herausgesucht haben, wollen wir noch Gäste dazu einladen. Unser Bekanntenkreis ist nicht gerade klein. Für einen Spielfilm lohnt eine weite Anreise nicht und so sollen nur die Freunde aus dem Ort kommen. Wir klicken den Schalter »Filter« auf der Steuertafel an. Das Programm löscht die Bildschirmmaske und öffnet eine weitere Steuertafel. Durch den Filter läßt sich der Datenbestand auf Sätze mit einem bestimmten Merkmal reduzieren (auch Selektion genannt). Für unseren Zweck positionieren wir den Cursor im Feld PLZ und geben »= 8000« ein. Wir könnten mit dem Ausdruck »LIKE [a-n]*« im Feld Namen noch den Kreis der Gäste auf jene beschränken, deren Nachname mit den Buchstaben der Folge A bis N beginnt. Doch wir bleiben auf dem Teppich. Ein Anklicken von OK bestätigt unsere Wahl. DB 2000 fragt noch, ob die Selektierkriterien gespeichert werden sollen. Wir entscheiden uns dagegen. Datensätze, die den eingesetzten Merkmalen nicht entsprechen, bleiben weiter in der Datei, werden aber bei Datenbankfunktionen nicht mehr berücksichtigt.

Eine solche Funktion ist zum Beispiel die Druckausgabe. Nach Anwahl von »Ein-/Ausgabe Formular« erscheint die Frage »Keine Druckmaske vorhanden. Standardformular benutzen?« auf dem Bildschirm. Wir ignorieren die Meldung und klicken OK an. Für den Fall, daß vorher als Ausgabegerät Drucker aktiviert wurde, besitzen Sie innerhalb kurzer Zeit (je nachdem) eine Liste der gewünschten Person. Andernfalls sehen Sie dieselbe Ausgabe am Bildschirm.

In der Liste befinden sich allerdings noch eine Vielzahl doppelter und überflüssiger Daten. So finden wir in jedem Datensatz die gleiche Postleitzahl und den gleichen Ort. Da wir unsere Freunde telefonisch benachrichtigen wollen, ist auch die Angabe der Straße überflüssig. Derart uninteressante Daten können Sie bei der Druck- und Bildschirmausgabe »ausblenden«, indem Sie die entsprechenden Felder schließen. Geschlossene Felder werden zwar nicht mehr angezeigt, bei den Datenbankfunktionen aber weiterhin berücksichtigt.

Komplexe Abfragen

Bei größeren Dateien kann es ziemlich lange dauern, bis DB 2000 alle Sätze mit der Postleitzahl 8000 findet, denn dazu müssen alle Datensätze sequentiell durchsucht werden. Um diese Zeit zu verkürzen, lohnt sich die Anlage einer Indexdatei. Mit der Menüfunktion »Neu Index« öffnet das Programm einen Requester mit einer Tabelle der für die jeweilige Datei verfügbaren Feldnamen. Wenn wir dies für unser Beispiel mit dem Feld »plz« machen, enthält die Indexdatei eine sortierte Folge aller in der Datei vorkommenden Postleitzahlen mit einem Verweis auf die Position des dazugehörigen Satzes. Datenmanager können auf sortierte Folgen grundsätzlich schneller zugreifen. Da bei DB 2000 während der Dateiarbeit nicht unbedingt eine Indexdatei geöffnet sein muß, ist gegebenenfalls eine Re-Indexierung vor Auslösen einer Suchfunktion notwendig.

Gehörten die bisher beschriebenen Suchverfahren noch zum Standard einer Dateiverwaltung, so kommen wir jetzt zu professionelleren Methoden. Dazu muß allerdings ein kleiner Umweg über den Maskeneditor gemacht werden. Bei der Druckmaske (Ausgabeformular) können bis auf Linien und Rechtecke alle bei der Eingabemaske beschriebenen Elemente verwendet werden. Zusätzlich lassen sich Felder duplizieren, um entweder mehrbahnige Etiketten oder Feldinhalte mehrfach (für Serienbriefe) zu drucken.

DB 2000 kann Datensätze, die ein bestimmtes Merkmal besitzen, zu einer Gruppe zusammenfassen und auf die numerischen Felder dieser Teilmenge mathematische Funktionen anwenden. Beispiel: In den Sätzen einer Lagerdatei befinden sich die folgenden

Angaben: Name des Artikels, die Nummern für Warenbereich, Warengruppe und Warennummer sowie der Lagerbestand und der Preis des Artikels. Bild 2 zeigt schematisch die Definition für ein Ausgabeformular, mit dem das Programm sowohl für Warenbereich als auch Warengruppe die Summe der Warenwerte (Lagerbestand x Preis) ausgibt. Dafür müssen zunächst die Felder für den Drucksatz ausgewählt werden. In unserem Fall sind das die Nummer, der Name, Preis und Lagerbestand des Artikels. Nun möchten wir für alle Artikel mit derselben Gruppennummer eine Aufsummierung von Lagerbestand und Warenwert haben.

Einsatz von Grafik und Sound

Diese Aufsummierung muß also dann ausgegeben und damit eine Gruppe beendet werden, wenn sich der Inhalt des Feldes Warengruppe ändert. Mit der Definition sogenannter Bänder legt der Anwender fest, welches Feld den Gruppenwechsel auslöst und was zu Beginn und am Ende einer Gruppe ausgegeben beziehungsweise errechnet werden soll. So sind die Summenfelder nicht Teil des Datensatzes, sondern nur für die Ausgabe als sogenannte abgeleitete oder errechnete Felder definiert worden. Dasselbe gilt für das Automatikfeld Datum. In ähnlicher Form haben wir Band 2 für den Warenbereich, die Seitenüberschrift sowie Report-Titel und -Ende festgelegt.

Unsere fiktive Datenbank besitzt für solche Aufgaben noch ein mächtiges Leistungsmerkmal. DB 2000 kann bis zu acht Dateien gleichzeitig bearbeiten. So kann sich ein Feld, auf das man sich allgemein, in den Selektierkriterien, bei Bestimmung des Gruppenwechsels oder einer Formel in abgeleiteten Feldern, bezieht, in einer der eröffneten Dateien befinden. Ein praktisches Beispiel: In einer Verkaufsdatei befinden sich die Kundennummer, Artikelnummer und die verkaufte Menge eines Artikels. Für die Rechnungsschreibung benötigt DB 2000 noch die Kundendatei und die Artikeldatei mit Name und Preis des Artikels.

Lösung der Aufgabe: 1. Alle drei Dateien öffnen. 2. Die entsprechende Ausgabemaske erstellen oder laden. 3. Mit dem Filterausdruck »Artikeldatei. Artikelnummer = Verkaufsdatei.Artikelnummer AND Kundendatei.Kundennummer = Verkaufsdatei.Kundennummer« festlegen, daß zu den Nummern der Verkaufsdatei die richtigen Preise und Adressen gesucht werden. 4. Eine nach Kundennummer aufsteigende Sortierfolge der Verkaufsdatei einstellen und die Fakturierung starten. Eine Gruppe ist in diesem Fall eine Rechnung und das den Gruppenwechsel auslösende Feld die Kundennummer aus der Verkaufsdatei.

Die meisten Anwender des Amiga werden sich wohl kaum mit Kunden- oder Artikeldateien herumschlagen müssen. Eines begeistert aber fast alle: die Grafik- und Soundfähigkeiten dieses Computers. Ein Datenmanager auf dem Amiga muß deshalb Dateien mit Bildern oder Tonfolgen verwalten können. DB 2000 kann dies mit Hilfe externer Felder. Extern bedeutet in diesem Fall, daß die eigentlichen Daten nicht Bestandteil der Datendatei sind, sondern sich in IFF-Dateien befinden.

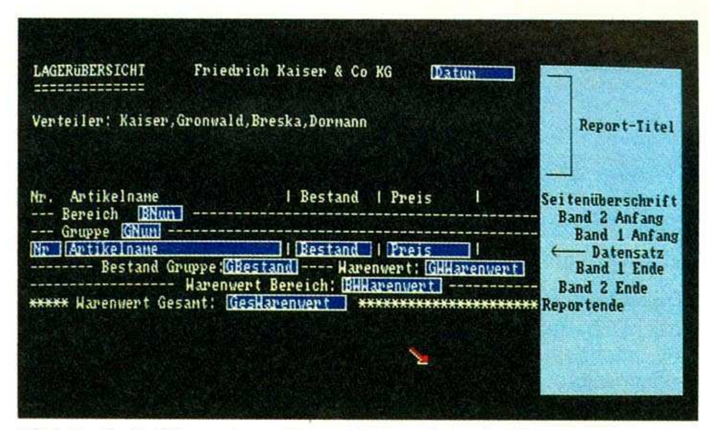


Bild 2. Definition einer Ausgabemaske mit DB 2000

Um die IFF-Dateien in die Datenbank zu integrieren, muß lediglich der Name der Datei in ein Feld vom Typ »Extern« eingetragen werden. Anklicken der Menüschalter Bildschau, Tonschau und Filmschau (de)-aktiviert die Wiedergabe von nur Grafik, nur Ton oder Grafik und Ton zusammen. Wenn Sie jetzt den schnellen Durchlauf einschalten, gibt DB 2000 den Inhalt der entsprechenden IFF-Dateien während des Durchblätterns wieder. Mit einer Veränderung der Anzeigegeschwindigkeit läßt sich die Dauer des Vortrags frei bestimmen. Natürlich können Sie sich auch mit den Datensatzschaltern manuell durch eine Filmschau hangeln. Praktische Anwendungen: Eine Sammlung mit digitalisierten Bildern heimischer Singvögel und deren Gesang, Produktdemo im Schaufenster. Die implementierte Hardcopyfunktion gibt die angezeigte Grafik auf den Drucker aus.

Sie kennen nun die meisten Funktionen von DB 2000. Eine Bedienung des Programms würde Ihnen wahrscheinlich nicht schwerfallen. Stellen Sie sich einmal vor, Sie haben mehrere Dateien im Speicher, die für Ihre Aufgabe richtigen Ein- und Ausgabemasken geladen, Filterkriterien entwickelt und ausgetestet, wollen gerade mit der Auswertung beginnen, da platzt Ihre Partnerin (oder ein Bekannter) in Ihren Hobbyraum und erinnert Sie an den für diese Zeit geplanten Kinobesuch. Also: Rechner ausschalten und am nächsten Abend die gesamte Startprozedur wiederholen.

Nicht mit DB 2000. Mit einem Klick auf »Projekt speichern« und der Angabe eines Dateinamens speichert das Programm Informationen über die augenblicklich verwendeten Dateien, Masken und die Systemeinstellungen. Geben Sie als Dateinamen »startup« ein, dann lädt DB 2000 diese Datei beim Programmstart und öffnet die Dateien, Masken und Filterkriterien automatisch. Ansonsten läßt sich mit »Objekt laden« dieser Mechanismus auslösen.

Zu den Systemeinstellungen gehören übrigens die Bildschirmfarben, Standardformate für Zahlen, Daten (Plural von Datum) und Zeit, Feld- und Satztrenner für die Ein- und Ausgabe der Daten in eine Textdatei und so weiter.

Sollte eine Datenbank unbedingt mit einer Programmiersprache ergänzt werden? Diese Frage wird kontrovers diskutiert. Unsere Antwort lautet: nein. Ein Programm mit flexibler Ausgabegestaltung kann eine Vielzahl von Problemen ohne eine einzige Programmanweisung lösen. Dies hilft dem Anwender, der zwar Fachwissen über das zu lösende Problem, aber nur wenig Computerkenntnisse besitzt. Wenn ein EDV-Profi allerdings die Aufgabe hat, einen Betrieb mit einer Datenverarbeitung auszurüsten, so wird er in der Regel ohne Programmiersprache nicht auskommen können, noch wollen. DB 2000 kann daher programmiert werden.

Die Datenbank Superbase

Die Merkmale von Superbase haben die Entwicklung von DB 2000 entscheidend beeinflußt. So sind die Auswertungsmöglichkeiten bei beiden Programmen fast gleich. DB 2000 ist allerdings einfacher zu bedienen.

Bei der Dateidefinition müssen Primär-, Sekundärattribute sowie Länge des Feldes festgelegt werden. Auch ein vom Standard abweichendes Format für Zahlen oder Daten sind an dieser Stelle einzustellen. Die Angabe von mindestens einem Indexfeld verlangt das Programm am Ende der Dateidefinition. Daher sind die Sätze immer nach dem Inhalt eines Feldes aufsteigend sortiert. Benutzerdefinierte Eingabeformulare unterscheiden sich nur durch die Position der Felder von der Eingabemaske. Superbase bietet also zur optischen Gestaltung der Eingabemaske weder ergänzende Texte noch grafische Elemente. Der Selektionsfilter besteht aus einer wenig praktikablen Anweisungszeile. Wurde bei DB 2000 das Selektionskriterium für die einzelnen Felder in das entsprechende Eingabefeld eingetragen/eingeklickt, so muß Superbase ein komplizierteres Kriterium mit UND, ODER und NOT verknüpfen (Beispiel: Name = "Ellen" AND Vorname = "Betrix").

Da nur etwa 40 Zeichen der gesamten Zeile zu sehen sind, kann das schon mal ein wenig unübersichtlich werden.

Ähnlich unkomfortabel ist die Ausgabe komplexer Auswertungen zu bewerkstelligen. Beginnen wir deshalb mit den einfacheren: Für die Darstellung der Daten auf dem Bildschirm besitzt Superbase einen Formular- und einen Listen-Modus. Dateneingabe ist nur im Formularmodus möglich. Mit der Feldanordnung

AMIGA SOFTWARE ZUM ABHEBEN

Word Perfect, engl.
DM 398,00

Update später in deutsch
DM 115,00

Digi View V 2.0

Neue PAL-Version mit deutschem Handbuch

Amiga 500/2000 DM 428,00

Amiga 1000 DM 368,00

SPIELE

A Mind Forever Voyaging	DM	89,00
Adventure Constr. Set	DM	79,00
Alien Fires	DM	89,00
Amegas	DM	49,00
Arazok's Tomb	DM	89,00
Archon II	DM	79,00
Arena	DM	79,00
Balance of Power	DM	89,00
Barbarien (Psygnosis)	DM	79,00
Bard's Tale	DM	97,00
Beat it	DM	29,95
Biotimer	DM	68,00
Black Couldron	DM	78,00
Borrowed Time	DM	79,00
Brataccas	DM	79,00
Ch. Basketb. Two on Two	DM	79,00
Championship Golf	DM	79,00
Chessmaster 2000	DM	88,50
Clever & Smart	DM	57,00
Crystal Hammer	DM	39,50
Deep Space	DM	88,50
Defenders of the Crown	DM	87,50
Dèjà Vu	DM	88,50
Faery Tale	DM	98,00
Feud	DM	29,95
Flightsimulator II	DM	118,00
Flip Flop	DM	72,00
Fußball Manager deut.	DM	72,00
Galileo	DM	215,00
Garrison II	DM	65,95
Goldrunner	DM	79,50
Grand Slam Tennis	DM	79,00
Guild of Thieves	DM	89,00
Hacker II	DM	69,00
Hardball	DM	78,00
Hollywood Hijinx	DM	89,00
Impact-Spiel	DM	49,50
Indoor Sports	DM	79,00
Jagd auf Roter Okt.	DM	72,00
Jewels of Darkness	DM	48,00
Jinxter	DM	84,00
Karate Kid II	DM	64,00
Karting Grand Prix	DM	29,95
Kings Quest II	DM	78,00
Leaderboard Golf	DM	74,90
Leaderboard Tournament	DM	47,00
Leviathan	DM	69,00
Little Computer People	DM	89,00
Lurking Horror	DM	86,00
Marble Madness	DM	68,00
Mind Breaker	DM	29,95
Mission Elevator	DM	59,00
Moebius	DM	78,00
Ninja Mission	DM	29,95
OCRE	DM	79,75
One on One	DM	98,00

SPIELE

Plutos	DM	59,95
Portal	DM	89,00
Powerplay	DM	59,95
Programm des Lebens	DM	155,00
Quintette	DM	99,00
S.Huey,Heli.Flight.Sim.	DM	67,00
Scenerys Disk Flightsim. II	DM	59,00
Shanghai	DM	79,00
Silicon Dreams	DM	69,00
Sinbad a. t.T. of t. Falcon	DM	87,00
Sky Fighter	DM	59,00
Soccer King	DM	29,00
Sorcerer	DM	99,00
Space Port	DM	59,00
Space Ranger	DM	29,95
Speed	DM	29,95
Starglider	DM	79,00
Strange New World	DM	48,00
Strip Poker	DM	79,00
Swooper	DM	48,00
Tass Times in Tonetown	DM	75,00
Terramex	DM	54,00
Terrorpods	DM	78,50
Testdrive	DM	89,00
Tetris	DM	59,95
The Big Deal	DM	78,00
The Final Trip	DM	29,95
The Pawn	DM	75,00
The Witness	DM	99,00
Tolteka	DM	58,00
Ultima IV	DM	99,00
Uninvited	DM	89,00
Vader	DM	29,95
Way of Little Dragon	DM	39,50
Western Games	DM	59,00
Winter Games	DM	64,00
Winter Olympiad 88	DM	57,00
World Games	DM	64,00

KOMMERZIELL

Analyze V 2.0	DM	349,00
BBS Bulletin Board	DM	210,00
Logic Works	DM	398,00
Logistix deutsch neue V	DM	329,00
OnLine!	DM	155,00
Organize!	DM	228,00
Page Setter PAL, Uml.	DM	349,00
Publisher 1000	DM	459,00
Scribble 2.0	DM	219,00
Superbase deutsch	DM	245,00
UBM-Text V 2.2 deutsch	DM	149,00
Vip Profesional engl.	DM	329,00
Vizawrite	DM	198,00
Word Perfect, engl.	DM	398,00

BÜCHER

Amiga 500 Schaltpläne	DM	60,00
Amiga 1000 Schaltpläne	DM	70,00
Amiga 2000 Schaltpläne	DM	110,00
Sidecar Schaltpläne	DM	40,00
Amiga 500 Buch M+T	DM	49,00
Amiga Assemb. Buch M+T	DM	59,00
Amiga Basic Data Becker	DM	59,00
Amiga C in Beispl. M+T	DM	69,00
Amiga DOS M+T	DM	59,00
Amiga DOS - Manual	DM	79,90
Amiga DOS Express	DM	79,90
Amiga Progr. Handbuch	DM	69,00
Deluxe Grafik m. d. A	DM	49,00
Grafik auf dem Amiga	DM	49,00
Hardware Ref. Manual	DM	62,50
Intuition Ref. Manual	DM	62,50
Komment. ROM-List. 1 d	DM	69,00
Komment. ROM-List. 2 d	DM	69,00
Progr. in Basic Franzis´	DM	48,00
Progr. m. A Basic M+T	DM	59,00
Progr. Praxis A-Basic M+T	DM	59,00
Progr. Praxis MS-Basic Tewi	DM	59,00
ROM-K. Libr. & Devices	DM	88,00
ROM-Kernel Ref.: exec	DM	62,50
Systempr. in C	DM	59,00

DEMO DISK

Aegis-Animator	DM	12,00
Aegis-Draw	DM	12,00
Digi-View (H.A.M.)	DM	12,00
Dynamic-CAD	DM	12,00
Logistix	DM	12,00
Modula M2 Amiga	DM	12,00
Perfect Sound	DM	12,00
Sculpt 3D (2 Disks)	DM	24,00
TV Text	DM	12,00
Vizawrite	DM	12,00
Zing!	DM	12,00
Zuma Fonts	DM	12,00

MUSIK

Audio Master	DM	98,00
Deluxe Musik Constr.	DM	229,00
Drum Studio	DM	69,00
Future Sound	DM	395,00
Instant Music	DM	79,00
It's just Rock'n Roll	DM	69,00
Midi-Interface	DM	99,00
Music Studio	DM	99,00
Pro Midi Studio	DM	349,00
Sonix	DM	145,00
Sound Sampler	DM	219,00

Wir senden tägl. ab Großversandlager.

SPRACHEN UND TOOLS

AC Basic	DM	412,00
CLI mate	DM	78,00
Devpac Assembler	DM	148,00
Fortran 77	DM	649,00
Gizmoz V 2.0	DM	155,00
Grabbit	DM	69,00
Lattice C Compiler V 4.0	DM	399,00
M2 Amiga Modula 2	DM	340,00
Manx Aztec C Profess. V 3.6	DM	399,00
Manx Aztec C Devel. V 3.6	DM	599,00
MCC Macro Assembler	DM	179,00
MCC Pascal	DM	248,00
MCC Shell	DM	118,00
MCC Toolkit	DM	99,00
Modula-2 Commercial	DM	619,00
Multi Forth - 83	DM	578,00
True Basic Interpreter	DM	349,00
Zing ! (CLI deluxe)	DM	179,00

GRAFIK

Aegis-Animator + Images	DM	249,00
Aegis-Draw	DM	199,00
Aegis-Draw plus	DM	385,00
Aegis-Images	DM	69,00
Aegis-Impact	DM	149,00
Art & Utility Disk 1	DM	65,00
Art Disk Print I	DM	65,00
Art Parts Volume 2	DM	68,00
Butcher V 2.0 PAL, deut.	DM	89,00
Calligrapher	DM	218,00
Deluxe Paint II PAL, deut.	DM	198,00
Deluxe Print	DM	139,00
Deluxe Video V 1.2	DM	228,00
Digi Paint PAL, deut.	DM	152,00
Digi View A1000 PAL, deut.	DM	368,00
Digi View A500/2000 PAL, d.	DM	428,00
Disney Animator	DM	595,00
Dynamic CAD	DM	998,00
Forms in Flight	DM	165,00
Gender-Changer f DigiView	DM	80,00
Laserscript f. Page Setter	DM	99,00
Newsletter Fonts	DM	49,00
Page Setter PAL, Umlaute	DM	349,00
Prism	DM	159,00
Sculpt 3D	DM	178,00
Silver	DM	264,50
Studio Fonts	DM	59,00
TV-Text 3D	DM	245,00
Videoscape 3D PAL, deut.	DM	349,00
Video Titler PAL	DM	229,00
X - CAD Designer	DM	995,00
Zuma Fonts 1	DM	69,00
Zuma Fonts 2	DM	69,00
Zuma Fonts 3	DM	69,00

Preisänderungen vorbehalten.



☐ Bitte schicken Sie mir kostenlos Ihren 44 seitigen Amiga-Software-Katalog

☐ Ich zahle per: beiliegendem Verrechnungsscheck

☐ Nachnahme(zuzügl. N.N-Gebühren)

zuzügl. Versandkosten von DM 6,50

Hiermit bestelle ich:



Name

Vorname

Straße

PLZ/Ort

Jumbo Soft Software Verlag GmbH Horemansstr.3, 8000 München 19 Tel. 089/1 23 40 65

beider Modi kann das Programm die Daten auf Bildschirm oder Drucker ausgeben. Der Anwender ändert so durch Umpositionieren der Felder in der Eingabemaske auch die Ausgabe-(formular)maske. Dasselbe gilt für die Ausgabeliste, denn auch diese läßt sich durch eine Veränderung der Feldlänge bequem am Bildschirm modifizieren.

anzustellenden Berechnungen fest. Ein einfaches Beispiel: REPORT COUNT SUM Beitrag+Spende GROUP Ort SUM Beitrag+Spende. Ergänzt durch die obige Felderanweisung enthält die ausgegebene Liste folgende Angaben: Eine nach Ortsgruppen sortierte Namensliste von Vereinsmitgliedern, deren Beitrag, Einkünfte der Ortsgruppen und des gesamten Vereins sowie die Anzahl der Datensätze (Mitglieder). Mit der zusätzlichen Eintragung »ASCENDING Ort« in der Anweisungszeile »Reihe« ordnet das Programm die Ortsgruppen in aufsteigender Reihenfolge an. Die Implementation eines zusätzlichen Filters im Abfragesystems offenbart wenig Logik. Besitzt Superbase doch bereits die bei der Selektion von Datensätzen verwendete Filterzeile.



Bild 3. Das Abfrage-Kommunikationsfenster von Superbase

Derart schnelle, aber auch einfache Ausgabeformatierungen sind sicher ausreichend für einfache Dateiverwaltungen, aber nicht für eine relationale Datenbank. Das dachten sich auch die Entwickler von Superbase und ergänzten das Programm mit einem zwar effektiven, aber leider kompliziert zu bedienenden Abfragesystem.

Zum Report vortreten

Das Abfragesystem besteht aus den fünf Anweisungszeilen Titel, Felder, Report, Filter und Reihe (Bild 3). In der Titelzeile kann ein von Superbase beim Ausdruck zentrierter Titel angegeben werden. Zwei Symbolschalter ergänzen diesen mit Datum und Seitennumerierung links und rechts vom Titel. Die einklickbare Kommandofolge in der Felderzeile legt die auszugebenden Felder fest. Einfaches Beispiel: »@5&20 Name,@30&20Ort,@60&8Beitrag+Spende AS "Summe"« Menge,Preis sind zwei Feldnamen. Das Zeichen <@> legt die Position und das Zeichen <&> die Länge in der Zeile fest. »Beitrag+Spende« ist ein abgeleitetes Feld mit der Spaltenüberschrift Summe.

Die Reportzeile legt das die Gruppenwechsel verursachende Felder und die mit allen selektierten oder einer Gruppe zugehörigen Sätzen

Die Abfrage von Superbase ist das derzeit leistungsstärkste Element eines auf dem Amiga verfügbaren Datenmanagers. Für den privaten Anwender sind solche Funktionen allerdings nur selten praktikabel anzuwenden. Aber bereits bei geschäftlichen Anwendungen kleinerer Art ist eine relationale Datenbank sinnvoll einsetzbar.

Wir haben den Bedienungskomfort von Superbase sicherlich nicht in den höchsten Tönen gelobt. Zweifelsohne ist das Programm die am einfachsten zu bedienende Datenbank für den Amiga. Dennoch bleibt sie hinter den Möglichkeiten des Amiga zurück. So wird mancher Anwender, erst recht nach ergebnislosen Versuchen, den Ausführungen im Handbuch zu folgen, für die Lösung seines Problems nach einer Programmiersprache rufen, bevor er erkannt hat, daß es mit diesem Programm auch ohne geht. Superbase kann leider nur Grafik-IFF-Dateien (Bild 4) verwalten.

Ebenso maßgeblich wie Superbase hat der »Datamat« von Data Becker die Entwicklung von DB 2000 beeinflußt. Dies gilt besonders für den Maskeneditor, das flexible Datenformat und die Textfunktionen des Programms.

Die Dateiverwaltungen

Neben den bei der Erläuterung von DB 2000 erwähnten Maskenfunktionen kann Datamat zusätzlich Kreise ziehen und diese sowie Rechtecke mit verschiedenen Mustern füllen. Leider übernimmt das Programm nur die bei der ersten Definition angelegten Felder in die Masken. Dadurch müssen nachträglich hinzugefügte Felder in den jeweiligen Formularen (Drucker und Eingabe) ein weiteres Mal angelegt werden.

Oder-Verknüpfung erzielen. Beispiel: MeierIMeyerIMaier (I = oder). Beim Selektieren gemachte Fehler können nur durch Aufheben der gesamten Selektion mit anschließender Neueingabe korrigiert werden. Mit der Definition eines Index sortiert Datamat die Datei nach dem Inhalt des Indexfeldes. Leider ist eine Speicherung der Indizes nicht vorgesehen. Beim Schließen der Datei gehen die Indextabellen verloren und müssen daher bei erneutem Öffnen der Datei wieder angelegt werden. Indexierung und Selektion schließen einander aus. Eine mehrfache Sortierung läßt sich durch wiederholte Sortierung selektierter Datensätze erzielen.

»GoAmiga! Datei« ist die deutsche Version des englischen »Softwood File«. Die Entwickler von Softwood nutzten ausschließlich das nur wenig Komfort bietende Konzept der sequentiellen Datei. Eine



Bild 4. Verwalten Sie Ihre IFF-Bilder mit Superbase

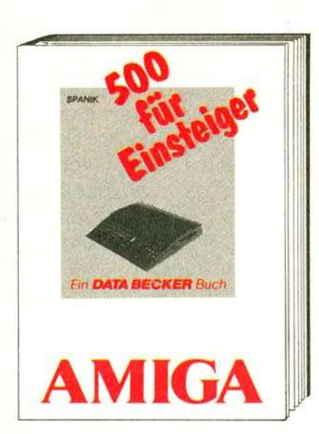
Datamat kennt nicht die Rechenfelder unserer fiktiven Datenbank. Allerdings wird ein in numerische Felder eingegebener einfacher Ausdruck — bestehend aus Konstanten, den Grundrechenarten und den 15 Rechenfunktionen (hauptsächlich Winkelfunktionen!) — sofort ausgewertet. Bei Aufruf der Menüfunktion »Selektion aktivieren« holt sich das Programm die Selektionskriterien aus der Eingabemaske und bildet die entsprechende Teilmenge. Damit verschwinden auch die Selektionskriterien auf Nimmerwiedersehen. Weitere Selektionen vermindern erneut die angezeigten Daten. Dadurch erzielt das Programm eine UND-Verknüpfung der angegebenen Selektionskriterien. Mehrere Kriterien in einem Feld lassen sich mit einer

Indexierung ist nicht vorgesehen. Die Datei wird beim Öffnen komplett ins RAM geladen, dort bearbeitet und anschließend wieder gepeichert. Intern lassen sich Datensätze zwar schnell bearbeiten oder suchen, aber der Dateigröße sind durch die Speicherkapazität in der Regel niedrige Grenzen gesetzt.

Die Definition der Felder ist umständlich. So muß für jedes neue zu definierende Feld die Menüfunktion »Feld erstellen« ausgelöst werden. Neben den Standardtypen besitzt das Programm die Typen »ja/nein« (erlaubte Eingaben nur ja oder nein) und »A-Z« (nur numerisch). Bei DB 2000 lösen die Überprüfungsfelder die Aufgabe dieser Datentypen.

GoAmiga! Datei kann Datensätze nach Inhalt eines Feldes

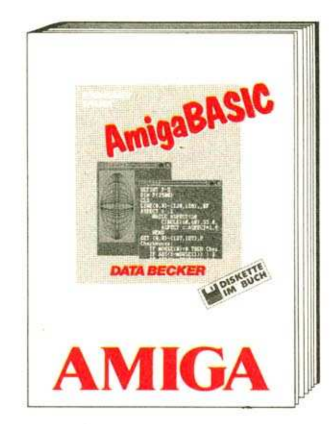
AMIGA BUCHHITS



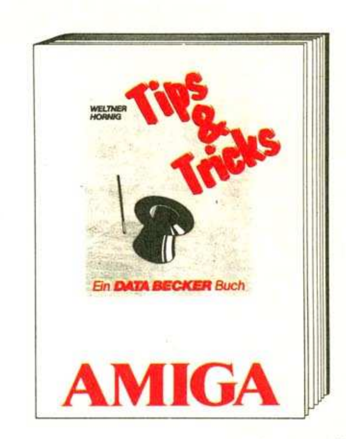
Wählen Sie gleich den richtigen Einstieg zu Ihrem Amiga 500. Denn das Handbuch läßt Sie dabei völlig allein. Versuchen Sie es lieber gleich mit Amiga 500 für Einsteiger. Hier heißt es: anschließen und loslegen. Verständlich für jedermann zeigt Ihnen dieses Buch: Workbench, AmigaBASIC, CLI und AmigaDOS. Locker aufbereitet bietet es Ihnen alles Wissenswerte. Bis hin zu den beim Amiga 500 mitgelieferten Zusatzprogrammen.
Amiga 500 für Einsteiger
343 Seiten, DM 39,–



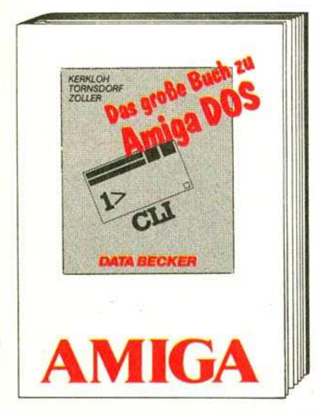
Läßt das Handbuch Sie auch in so manchen Dingen allein, das große Amiga-2000-Buch nicht. Hier finden Sie eine umfassende Einführung in die Arbeit mit Ihrem neuen Rechner – und mehr als das. Sind Sie erst einmal mit dem Amiga 2000 „per Du", zeigen Ihnen die Autoren, was einen Amiga-Profi ausmacht: Kickstart im RAM, PC-Audioausgänge, erste Hilfe bei Harddisk-Abstürzen, Laufwerkeinbau in den Amiga 2000 und, und, und. Sollten Sie also noch Fragen zu Ihrem Rechner haben, hier finden Sie die Antworten.
Das große Amiga-2000-Buch
Hardcover, 684 Seiten, DM 59,–



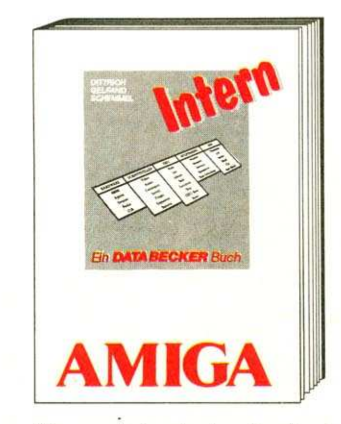
Das erfolgreiche Buch zu Amiga-BASIC. Erweitert um Kickstart 1.2, neuer Workbench und Amiga 500 & 2000. Alles, was BASIC-Programmierern Spaß macht: Grafik und Sound, Laden und Speichern von Graficraft-Bildern in BASIC-Programme, sequentielle und relative Dateien, Business-Grafik, Computeranimation, Windows, Umgang mit IFF-Bildern, Sprachausgabe und, und, und. Das Buch für Einsteiger, Aufsteiger und Profis.
AmigaBASIC
Hardcover, 774 Seiten, DM 59,–



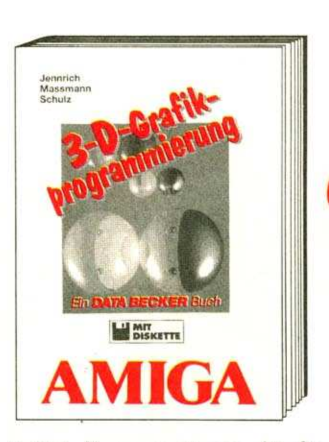
Amiga Tips & Tricks. Ein Buch, das voller Überraschungen steckt: Gestaltung eigener Programme, Tips & Tricks zum AmigaBASIC, Einbinden von Maschinenprogrammen in AmigaBASIC, Einsatz von DOS-Routinen, optimierende Hilfsprogramme für AmigaBASIC-Programme, Tips zur Arbeit mit der Workbench, Aufbau der Icons. Mit vielen Anregungen, aber auch fertigen Lösungen. Ein Griff in die Trickkiste, und ungeahnte Möglichkeiten tun sich auf.
Amiga Tips & Tricks
Hardcover, 473 Seiten, DM 49,–



Hier erfahren Sie alles, was Sie bei Ihrer praktischen Arbeit mit dem AmigaDOS wissen sollten: Umlenken der Ein- und Ausgabe, sinnvoller Einsatz des Jokers, Arbeiten mit RAM-Disk und CLI, Batch-Dateien, STARTUP-Sequenz, Multitasking mit dem CLI, Aufbau der CLI-Befehle, Programmierung eigener CLI-Befehle, neue CLI-Befehle in BASIC und C. Dazu ein ausführlicher, gut strukturierter Nachschlageteil. Wer also mit dem AmigaDOS arbeitet, sollte dieses Buch in greifbarer Nähe haben.
Das große Buch zu AmigaDOS
Hardcover, 310 Seiten, DM 49,–



Alles zum Amiga: In einem Band und absolut auf dem neuesten Stand: 68000-Prozessor, CIA, Blitter, Customchips, die wichtigsten Strukturen von EXE, I/O-Handhabung, Verwaltung der Resources, Multitasking, EXEC-Base, resetfeste Programme, DOS-Funktionen, IFF-Format, Programmstart von CLI und Workbench, Programmierung der EXEC- und DOS-Routinen und, und, und. Eben ein typischer Intern-Band, in dem wieder einmal nichts Wissenswertes fehlt.
Amiga Intern
Hardcover, 639 Seiten, DM 69,–



3-D-Grafikprogrammierung – hier finden Sie Grafikalgorithmen für absolut realistisch gestaltete Bilder. Die einzelnen Vorlagen lassen sich dabei mit einem Editor problemlos eingeben und solange durch die Wahl verschiedener Materialien, Farben, Lichtquellen und Spiegelungen verfeinern, bis Sie eine absolut naturgetreue Darstellung erreicht haben.
Amiga 3-D-Grafikprogrammierung
Hardcover, 283 Seiten
inkl. Diskette, DM 59,–

Super

Regelmäßig in der DATA WELT: Amiga Window – das Forum für den engagierten Amiga-Anwender. Mit kreativen Projekten, Interviews, Software-Tests und wichtigen News. Und: Auch was sich sonst so in der Computerszene tut, erfährt der Amiga-Anwender. Die DATA WELT – das aktuelle Computermagazin. Monat für Monat überall da, wo es Zeitschriften gibt.

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
unabhängig von der bestellten Stückzahl
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

	DB 2000	Superbase	Datamat	GoAmiga! Datei	Organize!
Technische Daten					
Dateiorganisation	relational	relational	seq./indexseq.	sequentiell	seq./ indexseq.
Dateiverwaltung	dynamisch	dynamisch	dynamisch	dynamisch	statisch
Paßwortvergabe	ja	ja	ja	nein	nein
Max. Dateigröße	hoch	17 GByte	2 GByte	k.A.	k.A.
Max. Anzahl offene Dateien	unbeschränkt	unbeschränkt	8	1	1
Anzahl Indizes/Datei	genug	1-999	0-80	0	1
Max. Anzahl Datensätze	genug	16 Millionen	2 Giga	k.A.	4,2 Billion
Max. Satzlänge	genug	unbeschränkt	64 KByte	k.A.	k.A.
Max. Anzahl Felder/Satz	genug	unbeschränkt	unbeschränkt	k.A.	128/Datei
Max. Feldlänge	ab 8 KByte	255	32 KByte	k.A.	254
Projektverwaltung	ja	nein	nein	nein	ja
programmierbar	ja	nein	nein	nein	nein
Reorganisation	automatisch	ja	ja	nicht nötig	ja
Steuerfunktionen					
- Schaltsymbole/Bildschirm	genug	viel	viel	mittel	wenig
- Shortcuts/Menü	ja	ja	ja	ja	ja
- Funktionstasten	ja	nein	nein	nein	ja
Information Dateiaufbau	ja	nein	ja	nein	ja
Kalender	ja	nein	ja	ja	nein
Softwareschutz	nein	Dongle	Seriennummer	Seriennummer	nein
deutsche Dokumentation	ja	ja	ja	ja	nein
deutsche Bed.-Führung	ja	ja	ja	ja	nein/ k. Umlaute
Dateidefinition					
Datentypen					
- Text/numerisch/Datum	ja	ja	ja	ja	ja
- Zeit	ja	nein	ja	ja	nein
- Extern	Grafik/Ton	Grafik	Grafik/Ton	Grafik/Ton	ja
Sekundärattribute					
- Überprüfung	ja	ja	ja	teilweise	Yes/No
- Repetier	Einzelfeld	nein	Einzelfeld	globaler Modus	nein
- Automatik	ja	nein	ja	nein	nein
- Rechenfeld	ja	ja	bedingt	ja	ja
- Pflichtfeld	ja	nein	ja	nein	nein
Datenformate					
- numerisch	genug	genug	genug	die wichtigsten	linksbündig
- Text	bündig, zentriert	keine	keine	bündig,zentriert	1
- Datum	genug	genug	wie DB2000	genug	1
- Zeit	genug	keine	wie DB2000	1	0
Definition änderbar	ja	ja	ja	ja	ja
Maskenfunktionen					
Anzahl Masken					
- Eingabe	beliebig	1	beliebig	1	beliebig
- Ausgabe	beliebig	Liste,Formular	beliebig	Liste,Formular	beliebig
Text	ja	nur Feldname	ja	ja	
Grafikelemente	Linie,Rechteck, Füllfarbe	keine	Linie,Rechteck Kreis,Füllmuster/ Flächen	keine	keine
Raster	ja	nein	ja	nein	nein
Größe					
- Eingabe	genug	273 Spalten	5000 x 5000 Pixel	k.A.	255 x 78
- Ausgabe	variabel	variabel	1000 x 255 Zeichen	k.A.	255 x 78
Neu-/Umdefiniton Felder	ja	nein	ja	nein	nur Länge
Übernahme der Felder					
- Maske -> Datei	wahlweise	%	nein	%	%
- Datei -> Maske	ja	ja	nein	%	%
Felder verschiebbar	ja	ja	ja	ja	ja
Felder vergrößerbar	ja	nein	ja	ja	ja
Feldduplikate					
- gleicher Inhalt	ja	prinzipiell	ja	nein	nein
- Inhalt nächster Satz	ja	nein	nein	nein	nein
Dateneingabe					
aus Textdatei (Import)	ja	ja	ja	ja	Zusatz- programm
durch Aktualisierung	ja	ja	nein	nein	nein
Textoperationen im Feld					
- Wordwrap	ja	nein	ja	nein	nein
- Editfunktionen	DEL,Backspace Cut,Copy,Paste	DEL,Backspace Cut,Copy,Paste	Del,Backspace	Del,Backspace	ja
- Zeichensätze	alle Amiga-Fonts	1	alle Amiga-Fonts	1	1
- Textattribute	fett,breit,kursiv unterstrichen	keine	fett,breit,kursiv unterstrichen	keine	keine
Kurztexte mit F-Tasten	ja	nein	ja	nein	nein
Elemente Rechenfeld/Filter					
Operatoren					
- mathematisch	5	4	%	4	5
- logisch	AND,OR,NOT	AND,OR,NOT	»und«,oder	»und«,NOT	AND,OR,NOT
- Vergleich	alle	alle	alle	alle	alle
- Verschachtelung (Klammer)	ja	ja	%	ja	ja
Funktionen					
- mathematisch	32	11	15	0	32
- Finanz/stat.	10	keine	0	0	3
- Text	10	10	0	0	0
- Datum	10	8	2	0	5
- Datei	10	CNT,SUM,AVG	CNT,SUM	0	SUM
Zahlenbereich	genug	bis 1.79E308	-10E-308 — 10E308	k.A.	16 Ziffern
Datumsbereich	Jahr 1-9999	Jahr 1-9999	julianisch	k.A.	1/1/1900- 31/12/1999
Währungssymbole	frei definierbar	18	frei definierbar	16	keines
Aktualisierung des Index	autom./Befehl	automatisch	autom./Index laden	%	autom./ Index laden

auf- oder absteigend sortieren. Ist Mehrfachsortierung eingeschaltet, so werden bei erneutem Sortieren nach einem anderen Feld nur noch diejenigen Sätze sortiert, dessen bei vorherigen Sortierläufen verwendeten Felder den gleichen Inhalt haben. So können die Sätze beispielsweise erst nach dem Ort, dann dem Nachnamen und schließlich nach dem Vornamen sortiert werden.

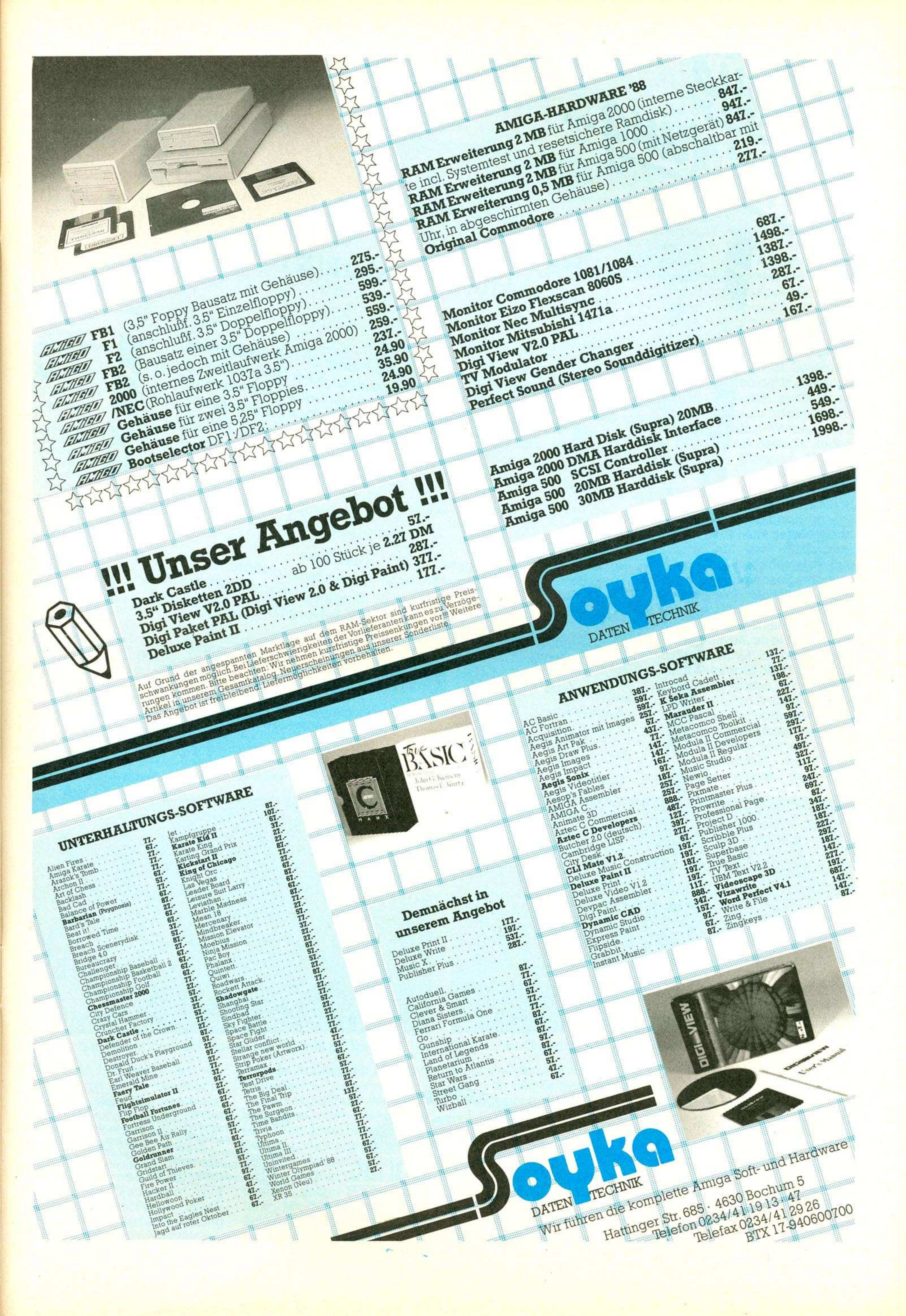
Durch Selektieren kann der Anwender eine Teilmenge der Sätze von der Verarbeitung/Anzeige ausschließen. Dies geschieht durch die Anwendung eines Selektierkriteriums (Beispiel: < = 4000) auf nur ein Feld (Bild 5). Ein Schalter bestimmt, ob die selektierten Sätze dem Kriterium entsprechen oder nicht (entspricht dem Operator NOT von DB 2000). Da keine Möglichkeit der Und-Verknüpfung beim Selektionskriterium besteht, läßt sich mit einer weiteren Funktion ein Bereich (von..bis) selektieren.

Formulare und Listen

Neben der Standardmaske kann der Anwender sich eigene Eingabeformulare erstellen. Innerhalb dieser Maske lassen sich die Felder beliebig positionieren sowie deren Länge einschließlich der Feldbezeichnung ändern. Beim Speichern der Eingabemaske sichert das Programm auch die Einstellungen für Druckerformat und festgelegte Sortierfolgen.

GoAmiga! Datei unterscheidet den Formular- und Listenmodus. Im Formularmodus können in die Eingabemaske Daten eingegeben oder angezeigt werden. Ein Mausklick in die Schaltsymbolleiste (außerhalb der Schaltsymbole) aktiviert auf einfache Weise den Listenmodus (Bild 5 Hintergrund). Auf das Ausgabegerät (Drucker oder Datei) gibt das Programm entweder das Eingabeformular oder aber die Liste aus. Sensible Daten lassen sich durch »unsichtbar machen« (DB 2000: Felder schließen) bei der Druckausgabe unterdrücken. Unverständlicherweise schwärzt das Programm derart ausgeschaltete Felder in der Liste des Bildschirms anstatt Platz für andere Spalten zu machen.

Wenn Sie eine einfache Dateiverwaltung für eine beschränkte Anzahl von Adressen, Rezepten, Vokabeln oder



AMIGA-HARDWARE '88
RAM Erweiterung 2 MB für Amiga 2000 (interne Steckkarte incl. Systemtest und resetsichere Ramdisk) 847.-
RAM Erweiterung 2 MB für Amiga 1000 947.-
RAM Erweiterung 2 MB für Amiga 500 (mit Netzgerät) 847.-
RAM Erweiterung 0,5 MB für Amiga 500 (abschaltbar mit Uhr, in abgeschirmten Gehäuse) 219.-
Original Commodore 277.-
AMIGA FB1 (3,5" Foppy Bausatz mit Gehäuse) 275.-
AMIGA F1 (anschlußf. 3,5" Einzelfloppy) 295.-
AMIGA F2 (anschlußf. 3,5" Doppelfloppy) 599.-
AMIGA FB2 (Bausatz einer 3,5" Doppelfloppy) 539.-
AMIGA FB2 (s. o. jedoch mit Gehäuse) 559.-
AMIGA 2000 (internes Zweitlaufwerk Amiga 2000) 259.-
AMIGA /NEC (Rohlaufwerk 1037a 3,5") 237.-
AMIGA Gehäuse für eine 3,5" Floppy 24.90
AMIGA Gehäuse für zwei 3,5" Floppies 35.90
AMIGA Gehäuse für eine 5,25" Floppy 24.90
AMIGA Bootselector DF1:/DF2: 19.90
Monitor Commodore 1081/1084 687.-
Monitor Eizo Flexscan 8060S 1498.-
Monitor Nec Multisync 1387.-
Monitor Mitsubishi 1471a 1398.-
Digi View V2.0 PAL 287.-
TV Modulator 67.-
Digi View Gender Changer 49.-
Perfect Sound (Stereo Sounddigitizer) 167.-
Amiga 2000 Hard Disk (Supra) 20MB 1398.-
Amiga 2000 DMA Harddisk Interface 449.-
Amiga 500 SCSI Controller 549.-
Amiga 500 20MB Harddisk (Supra) 1698.-
Amiga 500 30MB Harddisk (Supra) 1998.-
!!! Unser Angebot !!!
Dark Castle 57.-
3.5" Disketten 2DD ab 100 Stück je 2.27 DM
Digi View V2.0 PAL 287.-
Digi Paket PAL (Digi View 2.0 & Digi Paint) 377.-
Deluxe Paint II 177.-
Auf Grund der angespannten Marktlage auf dem RAM-Sektor sind kurzfristige Preisschwankungen möglich. Bei Lieferschwierigkeiten der Vorlieferanten kann es zu Verzögerungen kommen. Bitte beachten: Wir nehmen kurzfristige Preissenkungen vor!!! Weitere Artikel in unserem Gesamtkatalog. Neuerscheinungen aus unserer Sonderliste. Das Angebot ist freibleibend. Liefermöglichkeiten vorbehalten.
Soyka
DATEN TECHNIK
ANWENDUNGS-SOFTWARE
AC Basic 387.-
AC Fortran 597.-
Acquisition 597.-
Aegis Animator mit Images 257.-
Aegis Art Pak 57.-
Aegis Draw Plus 437.-
Aegis Images 77.-
Aegis Impact 147.-
Aegis Sonix 147.-
Aegis Videotitler 167.-
Aesop's Fables 97.-
AMIGA Assembler 187.-
AMIGA C 257.-
Animate 3D 257.-
Aztec C Commercial 888.-
Aztec C Developers 487.-
Butcher 2.0 (deutsch) 127.-
Cambridge LISP 397.-
City Desk 277.-
CLI Mate V1.2 67.-
Deluxe Music Construction 197.-
Deluxe Paint II 197.-
Deluxe Print 187.-
Deluxe Video V1.2 197.-
Devpac Assembler 197.-
Digi Paint 117.-
Dynamic CAD 888.-
Dynamic Studio 347.-
Express Paint 157.-
Flipside 97.-
Grabbit 67.-
Instant Music 87.-
Introcad 137.-
Keybord Cadett 77.-
K Seka Assembler 137.-
LPD Writer 198.-
Marauder II 67.-
MCC Pascal 227.-
Metacomco Shell 147.-
Metacomco Toolkit 97.-
Modula II Commercial 597.-
Modula II Developers 297.-
Modula II Regular 177.-
Music Studio 97.-
Newio 497.-
Page Setter 327.-
Pixmate 117.-
Printmaster Plus 97.-
Prowrite 247.-
Professional Page 697.-
Project D 87.-
Publisher 1000 347.-
Scribble Plus 187.-
Sculp 3D 187.-
Superbase 227.-
True Basic 297.-
TV Text 187.-
UBM Text V2.2 147.-
Videoscape 3D 277.-
Vizawrite 197.-
Word Perfect V4.1 687.-
Write & File 147.-
Zing 147.-
Zingkeys 87.-
True BASIC
John G. Kemeny
Thomas E. Kurtz
AZTEC
C
MANX
UNTERHALTUNGS-SOFTWARE
Alien Fires 77.-
Amiga Karate 67.-
Arazok's Tomb 77.-
Archon II 77.-
Art of Chess 77.-
Backlash 67.-
Bad Cad 57.-
Balance of Power 77.-
Barbarian (Psygnosis) 67.-
Bard's Tale 87.-
Beat it! 27.-
Borrowed Time 67.-
Breach 67.-
Breach Scenerydisk 37.-
Bridge 4.0 57.-
Bureaucrazy 87.-
Challenger 27.-
Championship Baseball 67.-
Championship Basketball 2 67.-
Championship Football 67.-
Championship Golf 67.-
Chessmaster 2000 77.-
City Defence 27.-
Crazy Cars 57.-
Crystal Hammer 37.-
Cruncher Factory 27.-
Dark Castle 57.-
Defender of the Crown 77.-
Demolition 27.-
Destroyer 87.-
Donald Duck's Playground 57.-
Dr. Fruit 27.-
Earl Weaver Baseball 97.-
Emerald Mine 27.-
Faery Tale 77.-
Feud 37.-
Flightsimulator II 97.-
Flip Flop 27.-
Football Fortunes 67.-
Fortress Underground 27.-
Garrison 67.-
Garrison II 67.-
Gee Bee Air Rally 67.-
Golden Path 57.-
Goldrunner 77.-
Grand Slam 87.-
Gridstart 27.-
Guild of Thieves 87.-
Fire Power 57.-
Hacker II 77.-
Hardball 97.-
Hellowoon 67.-
Hollywood Poker 47.-
Impact 47.-
Into the Eagles Nest 67.-
Jagd auf roter Oktober 67.-
Jet 87.-
Kampfgruppe 107.-
Karate Kid II 67.-
Karate King 37.-
Karting Grand Prix 27.-
Kickstart II 27.-
King of Chicago 87.-
Knight Orc 77.-
Las Vegas 27.-
Leader Board 67.-
Leisure Suit Larry 87.-
Leviathan 67.-
Marble Madness 67.-
Mean 18 87.-
Mercenary 77.-
Mindbreaker 27.-
Mission Elevator 57.-
Moebius 77.-
Ninja Mission 37.-
Pac Boy 27.-
Phalanx 27.-
Quintett 87.-
Quiwi 57.-
Roadwars 57.-
Rockett Attack 27.-
Shadowgate 87.-
Shanghai 77.-
Shooting Star 27.-
Sindbad 77.-
Sky Fighter 57.-
Space Battle 27.-
Space Fight 27.-
Star Glider 77.-
Stellar conflict 77.-
Strange new world 47.-
Strip Poker (Artworx) 77.-
Terramax 57.-
Terrorpods 67.-
Test Drive 67.-
Tetris 57.-
The Big Deal 77.-
The Final Trip 27.-
The Pawm 87.-
The Surgeon 137.-
Time Bandits 57.-
Trivia 27.-
Typhoon 67.-
Ultima 77.-
Ultima II 77.-
Ultima III 77.-
Uninvited 77.-
Wintergames 67.-
Winter Olympiad '88 57.-
World Games 67.-
Xenon (Neu) 57.-
XR 35 27.-
Demnächst in unserem Angebot
Deluxe Print II 177.-
Deluxe Write 197.-
Music X 537.-
Publisher Plus 287.-
Autoduell 87.-
California Games 77.-
Clever & Smart 67.-
Diana Sisters 57.-
Ferrari Formula One 77.-
Go 77.-
Gunship 87.-
International Karate 67.-
Land of Legends 87.-
Planetarium 97.-
Return to Atlantis 87.-
Star Wars 67.-
Street Gang 57.-
Turbo 47.-
Wizball 67.-
DIGI-VIEW
User's Manual
Soyka
DATEN TECHNIK
Wir führen die komplette Amiga Soft- und Hardware
Hattinger Str. 685 · 4630 Bochum 5
Telefon 0234/41 19 13 · 47
Telefax 0234/41 29 26
BTX 17-940600700

	DB 2000	Superbase	Datamat	GoAmiga! Datei	Organize!
Suchoperationen					
Max. Länge Filter	genug	255 Zeichen	unbeschränkt	k.A	255
relationale Verknüpfung	ja	ja	nein	nein	nein
Selektion	ja	ja	ja	ja	ja
Wildcards					
– Joker <*>	ja	ja	ja	ja	nein
– Bereich	ja	ja	ja	ja	ja
– belieb. Zeichen <?>	ja	ja	ja	nein	nein
– global	ja	nein	ja	nein	nein
Datenausgabe					
Ausgabeformatierung d. Befehle	nein	ja	teilweise	nein	nein
in Textdatei (Export)	ja	ja	ja	ja	ja
in Druckdatei	ja	ja	ja	ja	ja
Felder öffnen/schließen					
– global	ja	ja	nein	ja	ja
– für einzelne Masken	ja	nein	ja	%	ja
Etiketten	Ausgabemaske	ja	Ausgabemaske	ja	ja
Serienbrief	Ausgabemaske	schwer	ja	nein	nein
Formularmodus	Ausgabemaske	ja	Ausgabemaske	ja	Ausgabemaske
Listenmodus	Ausgabemaske	ja	ja/A.-Maske	ja	Ausgabemaske
in neue Datei					
– prinzipiell	ja	ja	ja	ja	mit Zusatzprogramm
– Generierung neuer Felder	ja	ja	nein	nein	nein
Umschalter Drucker/Bildsch.	ja	ja	ja	nein	ja
Hardcopy	ja	ja	ja	nein	nein
Gruppenwechsel					
– durch Befehl	nein	ja	unvollkommen	nein	nein
– durch Maskendefinition	ja	nein	nein	nein	nein
Sortierung					
– Sortierfeld	Index/beliebig	Index/beliebig	nur über Index	beliebig	nur über Index
– Genauigkeit	wählbar	nur 100%	wählbar	100%	100%
– mehrfach	ja	ja	umständlich	umständlich	nein
Ausgabe-Abbruch	ja	<Ctrl C>	Schalter	Mausklick	nein
Ausgabeformat					
– Etiketten	Ausgabemaske	Einstellpanel	Ausgabemaske	Eingabepanel	Ausgabemaske
– Seite	Ausgabemaske	nur Preferences	Eingabepanel	Eingabepanel	Eingabefeld
– Druck	Schriftart,-Dichte Zeilenabst.	nur Preferences	Schriftart,-Dichte Zeilenab. Blocksatz	Zeilenabstand	Leerzeilen
– Kopf-/Fußzeile	Ausgabemaske	Ausgabeanweisung	Eingabepanel	automatisch	automatisch
– Seitennummer	Ausgabemaskenfeld	automatisch	Steuerzeichen Kopfzeile	automatisch	nein
– Datum/Zeit	Ausgabemaskenfeld	autom./nein	Steuerzeichen Kopfzeile	nein	nein

Bemerkung: Die Angabe »genug« bedeutet, daß eine genaue Angabe eines Merkmals bei DB 2000 nicht praktikabel oder bei den anderen Programmen in ausreichender Form vorhanden ist.
»k.A.« = keine Angabe; »%« = Angabe nicht möglich, da Voraussetzungen für dieses Merkmal fehlen.
CNT, SUM, AVG = Satzzähler, Summe, Mittelwert

Tabelle. Die wichtigsten Merkmale der vorgestellten Datenmanager

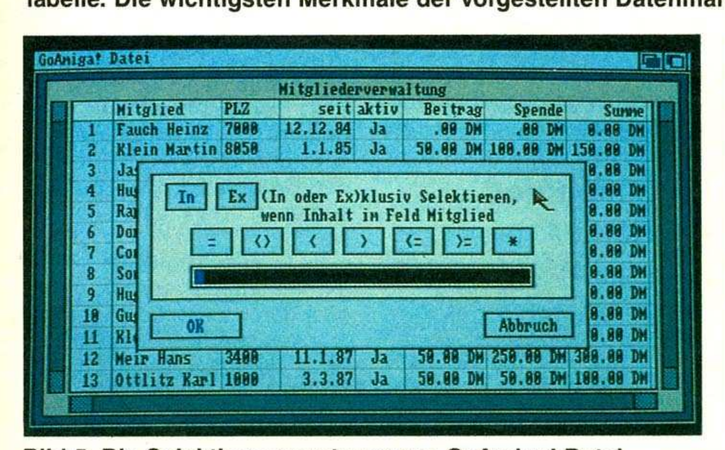


Bild 5. Die Selektionsoperatoren von GoAmiga! Datei

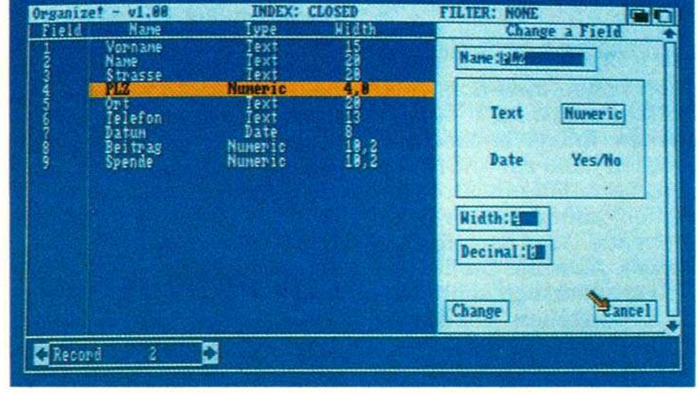


Bild 6. Mit Organize! die Datenstruktur festlegen

ähnliches benötigen, sich ungern durch unüberschaubare Funktionsvielfalt verwirren lassen und narrensicherer Bedienung bevorzugen, dann ist GoAmiga! Datei Ihr Programm.

Mit Organize! kommen wir zu unserem letzten Kandidaten. Dieses Programm wird neuerdings als Paket zusammen mit der Textverarbeitung Scribble! und der Tabellenkalkulation Analyze! für 398 Mark ausgeliefert.

Die Feldattribute von Organize! sind eher bescheiden. Das Programm bietet bei der Dateidefinition nur die Datentypen Text, Numeric, Date und Yes/No an (Bild 6). Also weder Grafik- noch Soundverwaltung. Mit der Einstellung der Feldlänge und der Anzahl Nachkommastellen sind die Möglichkeiten der Formatdefinition für alle Felder ebenfalls ausgeschöpft.

Das Programm nutzt für Eingabe und Ausgabe der Daten dieselbe Maske. Wie bei den anderen Programmen auch, läßt sich das Anwenderformular mit dem Maskeneditor individuell einstellen. Dazu gehört eine freie Positionierung des Feldes mit der Maus, Änderung der Feldanzeigelänge und des untrennbar mit dem Feld verbundenen Feldbezeichners. Texte lassen sich beliebig in der Maske positionieren, Felder öffnen und wieder schließen. Die wohl mächtigste Funktion des Programms ist jedoch die Definition von Rechenfeldern. Dafür, wie auch für vier Filterzeilen, stehen dem Anwender die Grundrechenarten einschließlich Exponentialoperator, drei Logikoperatoren, 32 mathematische, drei finanzmathematische und fünf Datumsfunktionen zur Verfügung. Von unserem Testquartett ist Organize! also am besten dafür geeignet, neben Textdaten auch lange Zahlenfolgen zu erfassen und statistisch auszuwerten.

Die Konditionallogik unterstützt dies (Beispiel: IF Feld < 0, Feld = 0, Feld = Feld). Hier macht sich die Verwandtschaft zur Tabellenkalkulation Analyze! bemerkbar.

Wenn Sie nicht gerade einen Report auf dem Bildschirm ausgeben, befindet sich das Programm im Anzeigemodus.

Mit zwei Schaltsymbolen können Sie nun schrittweise durch die Datei blättern. Unabhängig davon, ob eine Indextabelle geöffnet ist oder nicht, zeigt Organize! die Datensätze in diesem Modus in der Folge an, wie sie in der Datei gespeichert sind. Der vorletzte Eingabesatz ist damit tatsächlich vor dem aktuellen und mit einem Rückwärtsschritt leicht erreichbar. Würde das Programm ihn automatisch in die Indexfolge einreihen, wäre er mit Auslösen der Speicherfunktion in den Tiefen der Datei verschwunden und nur durch Aktivieren einer Suchfunktion wieder im Zugriff. Obwohl Organize! noch am wenigsten eine Projektfunktion benötigen würde, besitzt es eine solche. Mit dem Öffnen eines

Fortsetzung auf Seite 156

Drucken ist besser als Schlucken

Ich versuche den Seikosha SP1000 AS über die serielle Schnittstelle an meinen Amiga 2000 anzuschließen. Der Drucker liefert allerdings nur einen Teil des zu druckenden Textes. Stets verschluckt er ein paar Zeilen. Kein Treiber brachte eine Verbesserung. An einem anderen Computer arbeitet der Drucker normal. Was muß ich beim Amiga tun? STEFAN FREITAG
1 Berlin 30

Wie spät ist es?

Im Amiga 2000 ist laut Angaben des Herstellers eine akkugepufferte Uhr eingebaut. Doch wie kann ich sie stellen? In den beigelegten Handbüchern finde ich nur Hinweise, die gewünschte Uhrzeit mit den »Preferences« zu stellen. Doch nach einem Reset stimmt die Zeit nicht mehr.
JAN TCHINITCHIAN
6715 Lambsheim

Mit den Preferences stellen Sie nur die Systemuhr des Amiga. Bei jedem Reset aktualisiert der Amiga 2000 in der »Startup-Sequence« die Systemzeit mit der Uhrzeit der gepufferten Uhr durch den Befehl »SETCLOCK OPT LOAD«. Wenn Sie umgekehrt »SETCLOCK OPT SAVE« verwenden, überträgt der Amiga die Systemzeit an die gepufferte Uhr. (ub)

Antriebsschwäche

Ich habe einen Schneider Matrix-Printer NLQ 401 an meinen Amiga angeschlossen. Ich habe aber noch keinen vernünftigen Druckertreiber gefunden. Für Texte verwende ich den Epson-Treiber. Aber bei Grafik sind die Resultate schlecht. Dauernd entstehen Leerräume oder Grafik und Textzeichen werden vermischt. Wer hat einen Treiber für meinen Drucker?
DAVID B. HOUGHTON
Schweiz

Grafik für alle

Gibt es einen Weg, Grafiken, die mit Amiga-Basic erstellt wurden, in eine IFF-Datei umzuwandeln. Ich denke zum Beispiel an Grafiken, die der Raytracer aus der AMIGA generiert. Kann auch der umgekehrte Weg — von IFF ins Basic-Format — beschritten werden?
D. TAPPE
6229 Walluf 1

Auf der Extras-Diskette Version 1.2 zum Amiga 2000/500 finden Sie im Ordner »BasicDemos« zahlreiche Programme zum Umgang mit IFF-Bildern. Die Funktion der Programme ist in der Datei »Bitte lesen« dokumentiert. Mit »LoadILBM-SaveACBM« laden Sie ein beliebiges IFF-Bild aus einem Basic-Programm. »SaveILBM« speichert eine Bildschirmgrafik als IFF-Datei. (ub)

Laufwerk geht auf Speicherklau

Ich bin Besitzer eines Amiga 1000 und eines externen Laufwerks 1010. Nun brauche ich einen Rat: Das zweite Laufwerk benötigt etwa 20 KByte. Gerade dieser Speicher fehlt bei manchen Anwendungen und Spielen. Bisher mußte ich das Laufwerk bei ausgeschaltetem Computer abziehen. Doch ich möchte den Amiga nicht immer ausschalten. Hat jemand eine Schaltung parat, die dies vermeidet?
BUCK ROGER
Belgien

Links oder rechts

In meiner »Notepad-Version« funktioniert die Taste <DEL> nicht richtig. Sie bewirkt das gleiche wie <BACKSPACE>, das heißt löscht das Zeichen links vom Cursor. Ist das eine Besonderheit meines Notepads?
CHRISTOPH GRÜN
4804 Versmold

Es liegt nur am Notepad, daß <DEL> nicht so funktioniert, wie in anderen Editoren. Eine verbesserte Version existiert bisher nach unseren Kenntnissen noch nicht. (ub)

FRAGEN SIE

Wenn Sie Probleme mit dem Amiga, mit Peripheriegeräten oder mit der Software haben, stellen Sie Ihre Fragen ans Leserforum des AMIGA-Magazins. So können Sie mit anderen Lesern in Kontakt treten, die bereits eine Lösung gefunden haben.

Blick in die Sterne

Wer kennt ein Astronomieprogramm für den Amiga?
MICHAEL GOTTWALD
Ausgabe 2/88

Auf zwei PD-Disketten finden Sie gute Astronomie-Programme:
— Panorama 14a-StarChart1,0
— AmigaLibDisk80- AmiGazer
B. NINK
6000 Frankfurt

Vorfreude

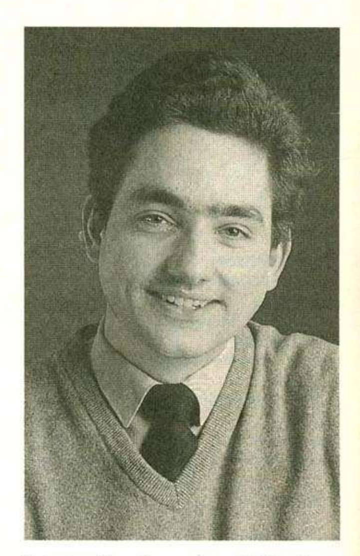
Zuerst ein dickes Lob an Euer Heft. Es ist einfach Spitze. In Kürze werde ich stolzer Besitzer eines Amiga 500 sein, und deshalb kaufe ich mir jetzt schon das AMIGA-Magazin. Nun habe ich eine Frage zu den Listings:
— Ist es nötig, die Zeilennummern einzutippen?
— Was ist mit den Buchstaben nach den Zahlen?
— Sind die Abstände zwischen Zeilennummer und Eingabebefehl nötig?
MARTIN AFFOLTER
Schweiz

Die Zeilennummern brauchen Sie nicht mit einzutippen. Aber die Zeichen nach den Zeilennummern sind wichtig. Diese Prüfzeichen geben Sie ein, wenn Sie mit dem Checksummer arbeiten (siehe AMIGA 3/88, Seite 62). »Checkie 42« kontrolliert anhand der Prüfzeichen, ob eine Zeile korrekt eingegeben wird. Die Abstände zwischen der Prüfsumme und dem Eingabebefehl müssen Sie berücksichtigen, sobald Sie drei Zeichen als Prüfsumme eingeben. (ub)

Unbekanntes Wesen

Was bedeutet die Guru-Nummer?
RALPH BENZINGER
6800 Mannheim

An der Nummer eines jeden Gurus erkennen Sie die Ursache für einen Systemabsturz (siehe auch AMIGA-Glossar, Ausgabe 11, Seite 76). Da die Systemmeldungen (»System-Alerts«) von großem Interesse für alle Amiga-Besitzer sind, werden wir in einer der nächsten Ausgaben eine komplette Liste veröffentlichen. Sie finden auch nähere Angaben zu den »Alerts« im Amiga-Programmierhandbuch von Markt und Technik, ISBN 3-89090-491-2, Preis 69 Mark. (ub)



Ich hoffe, Ihnen gefällt die Zusammenstellung des Leserforums.
Ihr Redakteur Ulli Brieden

Hals zu voll

Häufig erscheint beim Arbeiten mit Amiga-Basic die Meldung: »File already open«. Wie läßt sich dies vermeiden? DR. T.REICHELT
Ausgabe 12/87

Aufgrund der Multitasking-Fähigkeit des Amiga muß jedes einzelne Programm sorgfältig darauf achten, einmal beanspruchte Betriebsmittel — hier der Drucker — wieder freizugeben. Wird dies nicht korrekt erledigt, bleibt die Reservierung bestehen. Folgende Aufrufe werden mit »File already open« abgeblockt. Wenn Sie in Amiga-Basic den Drucker per LPRINT ansprechen, muß das Programm mit END verlassen werden. Der Befehl schließt den Druckerkanal ordnungsgemäß. Wenn Sie den Drucker mit PRINT# ansprechen, darf das entsprechende Statement CLOSE nicht fehlen. Sollte ein Programm durch <CTRL C> oder durch einen Fehler unterbrochen werden, geben Sie END beziehungsweise CLOSE über die Tastatur ein. Nur so lassen Sie das Betriebssystem nicht im unklaren.
Dr. PETER KITTEL
Commodore Deutschland
Sales Support Dokumentation

Digitizer

Mein Amiga 500 arbeitet nicht mit dem in der 68000er veröffentlichten Digitizer. Das Gerät läuft an einem 1000er meines Freundes einwandfrei. Ich habe die Platine bereits an die geänderte

Pinbelegung des A500 angepaßt. Dennoch kann ich im Adjust-Modus keinen Abgleich durchführen. Unabhängig von der Gleichspannungseinstellung an P1 bleibt der Bildschirm blau und springt nicht auf Schwarz um. Muß eventuell noch etwas geändert werden? ULRICH SCHMIDT 5460 Linz/Rh.

Aber bitte in Farbe

Nach langem Überlegen habe ich mich zum Kauf eines Amiga 500 entschlossen. Nun möchte ich ihn an einen Farbfernseher anschließen, der mit einem Scartstecker ausgerüstet ist. Was habe ich zu tun?

THOMAS SOMMERFELD
2350 Neumünster

In der AMIGA 10/87 auf Seite 36 finden Sie eine Anleitung für den Bau eines Kabels. (ub)

MS-DOS-Emulator

Zu der Anfrage im AMIGA-Magazin 2/88, inwieweit der MS-DOS-Emulator kompatibel zum IBM-PC ist, möchte ich ein paar Antworten geben:

Bisher läuft der MS-DOS-beziehungsweise IBM-Emulator 1.2 mit meinem Amiga 500 einwandfrei. Im Vergleich zum IBM-PC/AT ist er nur ein wenig langsam. Ich kann ohne weiteres Programme mit einem 3½- oder 5¼-Zoll-Laufwerk laden. Der Laufwerktyp muß in der Mountlist eingestellt werden. Wenn der Speicher ausreicht, laufen etwa 98 Prozent aller MS-DOS-Programme. Ich arbeite ziemlich viel mit dem Emulator. Nur ab und zu treten Schwierigkeiten mit der Tastaturbelegung auf. Wer den Emulator benutzt, muß natürlich aufpassen, daß er die richtige MS-DOS-Version besitzt. Diese wird nach dem Start des Emulators geladen.

LAURENT MUENSTER
Luxemburg

Task oder Aufgabe

Manche Artikel und Listings sind für Ausländer kaum zu lesen. Wann lernen die Deutschen mal vernünftig Englisch. Wann hören die Deutschen auf, einen Computer »Rechner«; einen Printer »Drucker« und ein File »Datei« zu nennen? Die Welt ist größer als Deutschland. Die AMIGA sollte ein gutes Vorbild geben.

M.L.P.M. VISSERS
Niederlande

Verdunklungsabsichten

Wie kann ich den gesamten Bildschirm, mit allen Fenstern und Screens, softwaremäßig abschalten? Ich möchte dies gerne wissen, weil der Amiga damit schneller würde, da die Custom-Chips nicht mehr auf den Speicher zugreifen brauchen. Am besten wäre eine Lösung in Assembler.

DANIEL GEMBIS
4600 Dortmund 30

PC ruft Amiga

Ich möchte ASCII-Texte mittels Diskette von meinem Amiga 1000 auf einen PC übertragen. Ich besitze bereits ein umschaltbares 5¼-Zoll-Laufwerk. Auf meiner Workbench befindet sich auch eine entsprechende »Mountlist«. Mit Hilfe dieser Liste kann ich das Laufwerk als IBM-formatiertes Laufwerk ansprechen. Wenn ich aber »Mount DF1:« eingebe, erscheint die Meldung: »Device DF1: allready mounted«. Deshalb meine Frage: Wie kann ein A1000 auf IBM-Disketten schreiben?

WOLFGANG PEST
8300 Ergolding

Nichts geht mehr

Nachdem ich mich längere Zeit intensiv mit der Hardware des Amiga beschäftigt habe, brennen mir zwei Fragen unter den Fingernägeln:

— Was passiert eigentlich mit den Daten, die an das Pseudo-Device »NIL:« geschickt werden? Woher weiß der Amiga beim Lesen von »NIL:«, wann er ein »EOF« senden muß?

— Kürzlich hatte ich einen Schreib-/Lesefehler; erstaunlicherweise auf der RAM-Disk. Das ist doch wohl ein dickes Ding. Da kann doch keiner der Speicherbausteine kaputtgehen, oder? A.TOM KRAFT 2000 Hamburg 90

Die Frage wohin die Daten gelangen, die an NIL: geschickt werden, hat in Fachkreisen bereits zu einer sehr angeregten Diskussion geführt (siehe auch Commodore Techn. Manual Vol. 1.4. »Amiga Devices: Details of datageneration and destruction in NIL:«). Auf den meisten Computern werden die Daten, die an das Pseudo-Device geschickt werden, in Hitze umgewandelt. Dies führt natürlich zu Kühlproblemen. Beim Amiga gingen die Entwickler einen anderen Weg. Laut Aussage des Handbuchs werden die Daten auf dem Amiga im Blitter gepuffert. Bis zu einer Million Bits kann er verarbeiten und dann in kleinen Portionen an den Copper übertragen. Dort erfolgt die Umwandlung in Wärme. Umgekehrt werden aus elektrischer Energie Bits im Blitter erzeugt, wenn Sie Daten aus NIL: lesen. Allerdings, und das ist das Geniale, schaut der Blitter zunächst in seinem BIT-Bugget nach, ob noch Bits von vorherigen Schreiboperationen im Puffer stehen. Auf diese Art spart der Amiga enorm viel Energie.

Die beschriebenen Schreib-/Lesefehler auf der RAM-Disk wird vermutlich jeder Programmierer bereits kennen. Die Ursache liegt in der zu geringen Rotationsgeschwindigkeit der RAM-Disk und somit an einer unsauberen Programmierung des RAM-Handlers. Per Interrupt steuert dieses Device die Geschwindigkeit, mit der die einzelnen Bits im Speicher verschoben werden. Hierzu sind die Assemblerbefehle ROL und ROR erforderlich. Aufgrund von Timing-Problemen kann nun häufig ein Befehl nicht rechtzeitig ausgeführt werden. Dadurch sinkt die Umlaufgeschwindigkeit der RAM-Disk und es kommt zu den beschriebenen Fehlern. Eine Abhilfe ist uns leider noch nicht bekannt, aber vielleicht weiß ein Leser Rat? (ub)

Laufwerk aus, Software läuft

Manche Programme vertragen kein zweites Laufwerk. Wie kann ich das Laufwerk »df1:« beim Amiga 2000 abschalten?

RONALD SCHLICHT
Ausgabe 2/88

Ein internes Zweitlaufwerk beim Amiga 2000 läßt sich abschalten. Bauen Sie statt des Jumpers »J36« einen einfachen Schalter ein. Der Jumper befindet sich beim Amiga 2000A rechts oben unter dem Netzteil-Schlitten (J301 beim 2000B) Achtung: Sie müssen den Schalter vor dem Einschalten des Computers in die gewünschte Position bringen. Ein nachträgliches Umstellen bringt nichts und sollte auf jeden Fall vermieden werden.

HEIKO JAHN
6480 Wächtersbach

Langsames Fast-RAM

Ich besitze einen Amiga 500. Nach Einbau der Speichererweiterung A501 mußte ich feststellen, daß bei PUT- und GET-Befehlen eine etwa 15prozentige Arbeitsverzögerung eintrat. Auch die Commodore-Hotline konnte mir nicht helfen. Ich erhielt die Auskunft, auf das neue CHIP-RAM könnten außer der CPU auch die drei Coprozessoren zugreifen. Dadurch bekäme die CPU nicht immer sofort ihre Daten. Nach dieser offenbar falschen Erklärung bekam ich den Rat, mich an eine Computerzeitschrift zu wenden.

Deshalb bitte ich Sie mir weiterzuhelfen oder meinen Brief zu veröffentlichen. Vielleicht kann mir ein Leser helfen. STEFAN THESEN

Beim Amiga 500 und 2000B ist der Speicher auf einer Erweiterung — immer noch als Fast-RAM bezeichnet — wenn er bei $C00000 beginnt, nicht schneller als das Chip-RAM. Der 68000-Prozessor kann bei einem Zugriff auf diesen Bereich tatsächlich von den Coprozessoren behindert werden. Beim Amiga 1000 ist dies ausgeschlossen. Dennoch können die Coprozessoren nur auf das normale Chip-RAM zugreifen. Der Grund für die besondere Behandlung des $C00000-RAM liegt an der geplanten Erweiterung des Chip-RAM auf 1 MByte. Liegt der RAM-Bereich einer Erweiterung dagegen nicht bei $C00000, verhält sich der Speicher wie das altbekannte Fast-RAM vom Amiga 1000 beziehungsweise Amiga 2000A. Dennoch bleibt die Frage, warum der GET-Befehl verzögert wird und nicht wenigstens genauso schnell abläuft wie ohne Speichererweiterung. Hierfür hat eventuell ein Leser eine Lösung. (ub)

ANTWORTEN SIE

Haben Sie schon eine Lösung zu einer der Fragen der Leser. Schicken Sie Ihre Antworten an das Leserforum, damit alle Leser von Ihrem Wissen profitieren. Umfangreiche Vorschläge werden wir eventuell auch in der Rubrik Tips und Tricks veröffentlichen.

IHRE MEINUNG

Schreiben Sie uns Ihre Meinung. Welches Problem brennt Ihnen unter den Fingernägeln?
— Schutz vor Viren,
— Software, die nicht läuft oder nur mit ganz bestimmten Gerätekonfigurationen funktioniert.
— Was halten Sie von der Praxis, Computerprogramme zu indizieren?
— Was gefällt Ihnen an der AMIGA (oder auch nicht)?

Ein Fest für alle

Mir ist vor kurzem der Begriff »Resetfeste RAM-Disk« zu Gehör gekommen. Wie kann ich eine solche soft- oder hardwaremäßig realisieren? THOMAS REIMANN

Die resetfeste RAM-Disk ist ein spezielle RAM-Disk, die auch durch Drücken von <CTRL A A> nicht gelöscht wird. Es handelt sich um eine reine Softwarelösung. Die ASDG-RAM-Disk auf der Public Domain-Disk »SACC 2« ist ein Beispiel. Die Installation erfolgt durch den Aufruf im CLI:

```
MOUNT vd0:
```

Hierzu ist ein entsprechender Eintrag in der Datei »devs/Mountlist« erforderlich. Die genaue Anleitung für die Initialisierung befindet sich auf der Diskette. Für Programmierer ist eine solche RAM-Disk eine große Hilfe. Sollte ein Absturz erfolgen, bleibt der Inhalt in der ASDG-Disk erhalten. Wichtige Software wie ein Compiler, ein Linker oder der Quellcode eines Programms sind dort gut aufgehoben. Nach jedem Reset sparen Sie die Zeit des Nachladens von Diskette. Außerdem sind Sie vor plötzlichen Angriffen der indischen Armee sicher (Guru-Meldung). (ub)

Adresse gesucht

Wo liegen beim Amiga 2000 neben den 512 KByte Chip-RAM die weiteren 512 KByte.

RALPH BENZINGER
6800 Mannheim 51

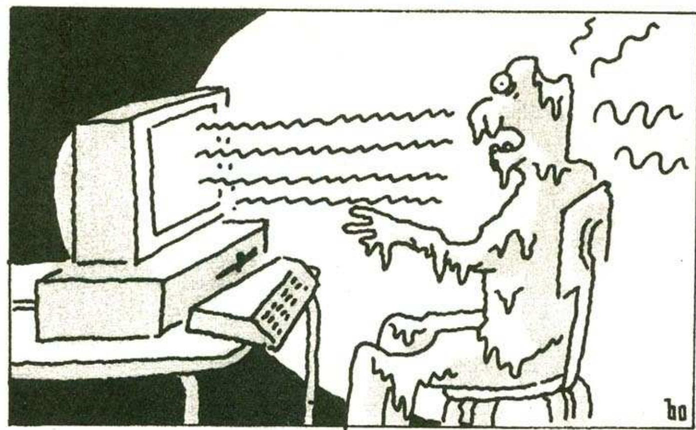
Der Speicherbereich der Erweiterung beginnt bei $C00000. Beim Amiga 2000B läßt sich die Startadresse mit einem Jumper verstellen. Dies wird erforderlich, wenn ein neuer Grafik-Chip mit erweitertem CHIP-RAM eingebaut wird. (ub)

Der Strahlen-Schutz

Ich finde die Zusammenstellung der Beiträge größtenteils gut ausgewogen. Nach meiner Meinung sollten Sie aber dem Umfeld der Beschäftigung mit Computern mehr Beachtung schenken. Darunter fällt zum Beispiel Computerkriminalität oder die gesundheitlichen Auswirkungen der Computerarbeit.

Hierzu eine Frage, die mir vielleicht ein Leser beantworten kann: Ich verbringe täglich etwa 1 bis 2 Stunden vor dem Amiga. Da der Monitor 1081 mit maximal 25 kV eine hohe Beschleunigungsspannung verwendet, würde mich das Ausmaß der ionisierenden Strahlung interessieren. Der Abstand zum Bildschirm beträgt 60 Zentimeter. Wie groß ist die Belastung? Läßt sie sich eventuell noch reduzieren?

MARTIN KATZENBEISSER
Wien



Strahlt der Bildschirm wirklich?

Bleibt bei Basic

Im Leserforum Ihrer Zeitschrift wird seit einiger Zeit über 'Basic oder C' diskutiert. Ich habe mit einigen Amiga-Besitzern über das Thema gesprochen und festgestellt, daß die meisten in erster Linie an Listings in Amiga-Basic interessiert sind. Hätte jeder einen kleinen C-Compiler, wäre es natürlich etwas anderes. Wäre es nicht wunderbar, wenn es nur eine Programmiersprache gäbe, die von allen eingesetzt wird? Jeder Anwendungsbereich würde mit dieser Sprache abgedeckt werden können. Programme wären auf allen Computern lauffähig. Die Realität sieht anders aus: Jeder Hersteller versucht durch den Einsatz von viel Geld und Geist das Gegenteil zu erreichen.

WERNER DOPPELSTEIN
1000 Berlin 20

2000er Tastatur für den Amiga 1000

Sie schreiben im AMIGA-Glossar, daß der Anschluß einer Amiga 2000-Tastatur nicht zu empfehlen sei. Da kann ich nicht zustimmen. Ich habe kürzlich eine 2000er-Tastatur am Amiga 1000 angeschlossen. Der Erwerb gestaltete sich allerdings schwierig. Bis auf einen Händler war niemand bereit mir ein einzelnes Keyboard zu liefern — auch Commodore nicht. Mit 250 Franken war ich dabei. Zum Anschluß an den Amiga 1000 mußte ich am 1000er-Kabel einen Stecker abschneiden und eine 5polige DIN-Buchse anlöten. Bastler sollten vorsichtig sein und die genaue Pinbelegung beachten. An die Buchse kann ich nun die Tastatur vom Amiga 2000 einstecken. Mit dem richtigen Tastaturtreiber funktioniert alles. Interessant ist der Anschluß für den Betrieb des Sidecars. Der Witz an der Sache: Die IBM-PC-Funktionstasten auf dem Zehner-Block können eingesetzt werden. Der einzige Nachteil: Die Tastatur läßt sich nicht unter den Amiga schieben.

MICHAEL BOSSHARD

Sicherheitslücken

Als ich in der Ausgabe 1/88 in der Rubrik »Aktuelles« den Artikel über deutsche Programmierer gelesen habe, war ich überrascht. Angeblich steht noch nicht fest, wann das Programm »Phalanx II« auf den Markt kommt. Seltsam, da mir das Spiel schon seit Ende November 1987 als Raubkopie vorliegt. Das sieht so aus, als gäbe es einige Sicherheitslücken bei der »neuen Zusammenarbeit« der deutschen Programmierer. Ich empfehle den Autoren einmal nachzuprüfen, ob Sie sich auf ihre Mitarbeiter voll verlassen können. Auf jeden Fall werde ich »Phalanx II« nicht mehr kopieren. Ich habe kein Interesse, daß mit den wirklich guten Programmierern ähnliches geschieht wie seinerzeit mit Lucasfilm Games. AMIGA SOLINGEN

Amiga 2000 jetzt ohne Streifen

Woher kommen die Streifen, die beim Amiga 2000 über den Bildschirm laufen? Gibt es Abhilfe?

JÜRGEN UMSTÄDTER
AUSGABE 11/87

Seit einer Woche habe ich beruflich mit dem Amiga 2000 zu tun. Da ich mich aber schon länger mit dem Computer beschäftige (lese das AMIGA-Magazin seit Juli 87), achtete ich sofort auf die zwei horizontalen Streifen. Jeweils im Abstand von etwa 15 Zentimetern laufen Sie von unten nach oben über den Bildschirm. Ich untersuchte das Netzteil auf eine Fehlerquelle — der Fehler war schnell gefunden. Der nicht genügend entstörte Lüfter entpuppte sich als Verursacher. Abhilfe schafft ein parallel zu den Zuleitungen eingelöteter Entstörkondensator (0,1 MicroFarad/250 Volt). Als Fernsehtechniker möchte ich jedoch darauf hinweisen, diesen Eingriff nur vom autorisierten Fachmann vornehmen zu lassen. BERND HARTMANN

IHRE ANWENDUNG

Auch das ist für die Leser von Interesse. Schreiben Sie ans Leserforum, wie Sie Ihren Amiga nutzen: Privat oder kommerziell? Wenn Sie meinen, eine besondere Anwendung gefunden zu haben — einschicken. Wir werden aus allen Einsendungen die interessantesten herauswählen und veröffentlichen. Bei allen Beiträgen zum Leserforum behalten wir uns allerdings vor, diese aus Platzgründen gekürzt wiederzugeben.

Die Alpen auf

Tolle Fraktalgrafiken in kurzer Zeit und mit höchstem Bedienungskomfort bietet »Fraktalberge«. Ein Muß für alle Fans von zufallserzeugter Grafik.

Der Amiga ist ein Computer mit Maus. Aber bei vielen Programmen wird dies leider nicht ausgenutzt. Die Eingabe von Zahlenwerten über die sogenannten Proportional-Gadgets (Schieberegler) ist sicherer und einfacher. Diese Regler werden bei »Fraktalberge« für viele Einstellungen benutzt und vereinfachen so die Bedienung sehr. Außerdem findet ein doppelter Schieberegler Verwendung, doch wie dieser aussieht, erfahren Sie etwas später.

Das Programm kann durch den Aufruf

`Fraktalberge`

von einem CLI-Fenster aus aktiviert werden. Sie können das Programm natürlich jederzeit mit »RUN Fraktalberge« starten, damit es auch im Hintergrund läuft. Nach dem Laden wird ein eigener Bildschirm geöffnet, auf dem zunächst nur die Menüzeile und ein eigenes Menü am unteren Rand zu sehen sind (siehe Bild). Zuerst wollen wir die drei Schalter in der Menüzeile des Bildschirms erklären.

-SAVE

Nach Anklicken des Schalters erscheint im oberen Teil des Fensters ein neues Fenster mit einem »CANCEL«- und einem Textgadget. Der Pfad und Name des Bildes können im Textgadget eingegeben werden. Nach Drücken von <RETURN> speichert das Programm dann das aktuelle Bild im IFF-Format. Das heißt, daß Sie die Bilder später mit Malprogrammen weiterverarbeiten können. Durch Anklicken des CANCEL-Schalters können Sie diesen Vorgang abbrechen. Beim Speichern des Bildes werden von der IFF-Routine einige Daten auf dem Textbildschirm ausgegeben. Diese sind jedoch für die Benutzung des Programmes vollkommen uninteressant.

AN

Dieses Gadget schaltet das Fenster ein, das sich im unteren Bildschirmabschnitt befindet. In diesem Fenster kann man alle Parameter verändern, das Programm beenden oder die Berechnung starten.

AUS

Dies ist das Gegenstück zu AN. Damit wird das Wahlfenster ausgeschaltet, falls man zum Beispiel das Bild speichern oder anschauen möchte. Das Fenster wird aber nicht ausgeschaltet, sondern nur in den Hintergrund gelegt. Dadurch ist es später wieder schneller auf dem Bildschirm zu bringen.

Das oben schon mehrfach erwähnte Wahlfenster enthält alle Schalter und Schieberegler, die man benötigt, um das Bild nach eigenen Wünschen zu gestalten.

Schieberegler 1-6

Mit diesen Reglern werden die Höhen der Stützpunkte verändert. Die Höhen variieren dabei zwischen einem Minimalwert von -30 und einem Maximalwert von 30. Die Mittelstellung entspricht daher der Höhe 0. Welche Nummer zu welchem Punkt gehört wird Ihnen sofort klar, wenn Sie die einzelnen Werte verändern und sofort eine Reaktion auf dem Bildschirm sehen. Die Punkte sind von links oben nach rechts und unten durchnumeriert. Probieren Sie es am besten gleich einmal aus. Nach der Veränderung eines Reglers erscheinen vier zusammenhängende Dreiecke auf dem Bildschirm. Die Höhe der einzelnen Eckpunkte ändert sich dabei je nach Einstellung.

Zahl der Iterationen 4-8

Mit diesen Schaltern kann durch einfaches Anklicken die Berechnung gestartet werden. Dabei gibt die Nummer des Knopfes die Tiefe der Berechnung an.

Die Berechnung der fraktalen Landschaft geschieht rekursiv. In der Tiefe »1« wäre nur ein Dreieck enthalten, was natürlich sehr langweilig wirkt. Für jede Erhöhung der Tiefe wird jedes vorhandene Dreieck in vier kleinere zerlegt. Um die Höhe eines Punktes zu erhalten, wird ein Mittelwert der schon vorhandenen Nachbarpunkte errechnet. Danach wird noch ein Zufallswert aufaddiert. Dieser Wert kann sowohl negativ als auch positiv sein und wird bei steigender Tiefe immer kleiner. Dadurch werden natürlich auch die Unterschiede geringer und das Bild wirkt realistischer.

Bei Tiefe 4 hat man also schon 64 (4^3) Dreiecke, die gezeichnet werden. Bei der höchsten Iterationszahl ergeben sich somit 4^7, also 4096 kleine Dreiecksflächen.

FAKTOR

Alle Höhen werden mit einem Faktor multipliziert. Je größer er ist, um so steiler und zerklüfteter wird die Landschaft. Von allzu kleinen Faktoren sei abgeraten, da die Landschaft dann eher wie Holland aussieht. Bei höheren Werten nähern Sie sich dann den Fjorden Norwegens.

RAND

Dieser Schalter verhält sich wie ein Flip-Flop-Schalter (zum Beispiel Einschaltknopf des Fernsehers). Je nach Stellung erscheint der Text »RAND AN« oder »RAND AUS«. Alle Dreiecke werden bei eingeschaltetem Rand von einem weißen Rahmen umgeben. Bei geringen Tiefen steigt dadurch der 3D-Effekt. Bei höheren Iterationstiefen sollten Sie darauf verzichten, da der tolle Eindruck des Bildes geschmälert wird. Der Text zeigt übrigens den Zustand an, den Sie erreichen, wenn Sie den Schalter betätigen. Zeigt es also »RAND AUS«, so ist der Randmodus im Augenblick eingeschaltet.

MEER

Schieber, der zur Einstellung der Meereshöhe verwendet wird. Dabei wird eine Änderung rechts neben dem Regler angezeigt. Die Mittelstellung entspricht auch hier wieder der Höhe 0. Alle Punkte, die unter Meereshöhe liegen, werden blau dargestellt und erwecken somit den Eindruck einer Wasserfläche.

ZUFALL

Die Landschaften werden jeweils von einer Zahl ausgehend generiert. Das heißt, daß zwei Landschaften mit demselben Startwert sich sehr ähnlich sehen. Mit dem Schieberegler können Sie Startwerte von 0 bis 99 vorgeben.

WINKEL

Die Landschaft kann unter einem Betrachtungswinkel, der zwischen 0 und 70 Grad

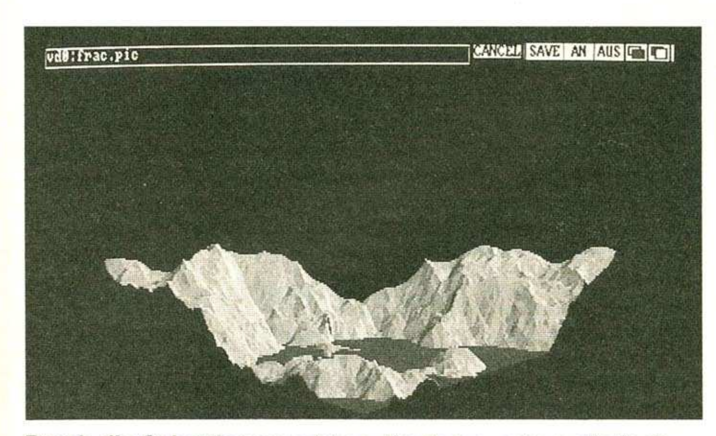
Jeden Monat 2000 DM GEWINNEN





Zu zweit geht alles besser, wie man an »Fraktalberge« sehen kann. Obwohl die beiden Autoren erst vor kurzem von Basic auf C umgestiegen sind, haben Sie doch ein sehr gutes, bedienerfreundliches Programm geschrieben. Der Umstieg erfolgte aus Geschwindigkeitsgründen und war wie wohl jeder sieht erfolgreich. Da beide Autoren noch zur Schule gehen, ist Ihnen der Gewinn von 2000 Mark ein willkommener Goldregen.

dem Bildschirm



Durch die Schattierung wirken die Berge sehr realistisch.

liegt, beobachtet werden. Der Winkel ist frei einstellbar. Bei einem Winkel von 0 Grad sehen Sie genau von vorne auf Ihre Berge. Am günstigsten sind Blockwinkel im Bereich von 15 bis 45 Grad. Wählt man den Winkel nämlich zu groß oder klein, wirkt das Bild nicht mehr so plastisch.

LICHT

Die zwei Gadgets für die Position der Sonne (oder auch Lampe) sind nicht beschriftet. Es handelt sich um einen vertikal stehenden Schieberegler zur Einstellung der Höhe und um den schon erwähnten doppelten Schieberegler am rechten Bildschirmrand. Mit ihm können Sie die X- und Y-Position der Sonne einstellen. Sie sehen diese Positon, als würden Sie von oben auf die Fläche herunterschauen. Auch hier findet man die optimale Einstellung am besten durch Ausprobieren. Stellen Sie die Sonne zu tief, wird Ihr Gebirge von unten beleuchtet, was nicht beabsichtigte Effekte herbeiführt. Im oberen Drittel erzielt man die besten Ergebnisse.

QUIT

Falls Sie das Programm verlassen wollen, genügt ein Klick auf diesem Schalter und schon befinden Sie sich wieder auf dem Workbench-Bildschirm. Da keine Sicherheitsabfrage durchgeführt wird, sollten Sie Ihr Bild vorher speichern, soweit dies gewünscht wird.

Zum Schluß noch ein paar allgemeine Bermerkungen zu Fraktalberge (siehe Listing).

Jedes Dreieck der fraktalen Landschaft ist schattiert. Der nötige Farbwert errechnet sich, indem man die Normale (die Senkrechte) auf die Fläche nimmt und den Winkel zwischen Lichtvektor und Normale ermittelt. Der erhaltene Wert wird mit den Werten der Graupalette multipliziert, wodurch ein winkelabhängiger Farbwert entsteht. Mit so einer Routine lassen sich für beliebige Flächen die passenden Farbwerte ausrechnen.

Zum Speichern des Bildes wird der IFF-Standard verwendet. Dabei werden zwei auf der Aztec-C-Compiler Diskette vorhandene Dateien (jiff.h und jiff__save.c) benutzt. Die zweite Datei muß in »jiff.c« umbenannt werden. Außerdem muß in der Headerdatei jiff.h der Ausdruck LORES durch HIRES ersetzt werden. Der Speicherroutine werden die folgenden Parameter übergeben:

— ein Zeiger auf den Bildnamen,
— ein Zeiger auf die Farbpalette,
— ein Zeiger auf die Bitmap,
— X- und Y-Offset,
— Breite des Bildes,
— Flag für komprimierte Speicherung.

Zur Darstellung der Farbtabelle werden die Farben aus den Registern gelesen und durch Bitverschiebung und einer AND-Verknüpfung zu Bytes gemacht, die den Farbanteilen der Farbe entsprechen.

Das Programm benötigt für ein Bild der Tiefe 5 zwölf Sekunden und bei Tiefe 8 ungefähr zehn Minuten. Durch Herunterziehen aller Bildschirme verkürzt sich diese Rechenzeit zirka um die Hälfte. Dabei kann die Berechnung nicht unterbrochen werden.

In etwa fünf Minuten haben Sie also die Alpen oder norwegische Fjorde auf Ihrem Bildschirm, die Natur hat da doch etwas länger gebraucht. Bloß mit dem Bergsteigen in den Fraktalbergen hapert es leider noch etwas.

(Chris Emmerich/
Tobias Dierig/rb)

Programmname:	Fraktalberge
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler	Aztec-C V3.4
Aufrufe:	cc fraktalberge -s ln fraktalberge.o -lm -lc

```
Programm : Fraktalberge

-----------

KhO /*FrakTal/Berge by Tobias Dierig & Chris Emmerich*/
5z  #include <exec/types.h>
dH  #include <functions.h>
DO  #include <stdio.h>
gf  #include <math.h>
dm  #include <graphics/gfxbase.h>
Vk  #include <intuition/intuition.h>
md  #include <graphics/gfx.h>
18  #include <graphics/gfxmacros.h>
hu  #include <graphics/rastport.h>
IN  #include <exec/memory.h>
UW  #include <graphics/regions.h>
3e  #include <graphics/gels.h>
18  #include <clib/macros.h>    /* für MAX */
5j  #include "jiff.c"  /* jiff.c und jiff.h in diesem DIR */
2u  /* defines für Gadget-ID's */
6j  #define k    0.017453292   /* pi/180 als Konstante */
J7  #define ST1  1             /* ID's für die ganzen Gadgets */
ZZ  #define ST2  ST1  + 1
eg  #define ST3  ST2  + 1
jn  #define ST4  ST3  + 1
ou  #define ST5  ST4  + 1
t1  #define ST6  ST5  + 1
U3  #define WI   ST6  + 1
es  #define ME   WI   + 1  /* immer +1 zum Einfügen von */
Zy  #define ZU   ME   + 1  /* zusätzlichen Gadgets */
Lq  #define FA   ZU   + 1
Bt  #define GR4  FA   + 1
bZ  #define GR5  GR4  + 1
gg  #define GR6  GR5  + 1
ln  #define GR7  GR6  + 1
qu  #define GR8  GR7  + 1
Ds  #define QU   GR8  + 1
Ti  #define BOU  QU   + 1
Wx  #define AN   BOU  + 1
yy  #define AUS  AN   + 1
iJ  #define SAVE AUS  + 1
Dn  #define GETT SAVE + 1
ak  #define CANC GETT + 1
kK  #define LI1  CANC + 1
fr  #define LI2  LI1  + 1
9c  struct IntuitionBase *IntuitionBase;
Ye  struct GfxBase       *GfxBase;
T6  struct Window        *MyWindow;
TN  struct Window        *MyWindow1;
sD  struct Window        *IconWindow;
ld  struct Window        *SaveWindow;
9j  struct Screen        *MyScreen;
5L  struct RastPort      *rp;
```

Listing. Norwegische Fjorde auf Knopfdruck in fünf Minuten mit »Fraktalberge«. Bitte mit dem Checksummer eingeben.

```
50 Ng  struct TmpRas          MyTmpRas;
51 Bg  struct AreaInfo        MyAreaInfo;
52 IR  struct IntuiMessage    *Message;
53 tH  struct Image           Imag1,Imag2,Imag3,Imag4,Imag5,Imag6;
54 9S  struct Image           Imag7,Imag8,Imag9,Imag10,Imag161,Imag16
       2;
55 rC  struct Gadget          AdrGad;
56 H7  struct PropInfo        AdrProp;
57 Ns  struct PropInfo        Prop1,Prop2,Prop3,Prop4,Prop5,Prop6;
58 dU  struct PropInfo        Prop7,Prop8,Prop9,Prop10,Prop161,Prop16
       2;
59 Zy  /* Unsere Variablen */
60 zz  PLANEPTR Pointer=0;  /* Pointer für TmpRas-Struktur */
61 70  float    winkel=15;  /* Anfangswinkel */
62 6X  float    maxl;       /* Lichtvektoren */
63 0o  float    l1=1;
64 zn  float    l2=0;
65 2r  float    l3=0;
66 OF  float    l,f,maxst,si,co;
67 cL  long     class,code; /* für Gadgetabfrage */
68 vW  int      Gadg,grad,grad1,meer,meer1,sy,sx,s,s1,vakt,vakt1,bo
       und,inh;
69 vp  int      kor[129][129],st[7],dec[3][3];  /* Koordinaten */
70 Ph  int      mittel(),expon();           /* eigene Funktionen */
71 1S  int      xmitte=320;      /* Mitte des Bildschirms */
72 CS  int      basis=185;     /* unterster Punkt des Fraktals */
73 zz  int      rr=1;
74 Tb  UBYTE    AreaBuffer[25];
75 GD  UBYTE    Stringbuff[61];
76 1D  UBYTE    Undobuff[61];
77 Dr  UWORD    ColorMap1;        /* für IFF-Routine Buffer */
78 2q  UBYTE    ColorMap2[48];
79 2m  char     string1[4],string2[4];    /* zum Darstellen der Wert
       e der Gadgets */
80 54  char     string3[4],string4[4];
81 cQ  /* Unser eigener Screen */
82 wu  struct NewScreen TheScreen =
83 Hk3    {
84 qE     0,0,640,256,4,0,1,           /* 640*400 mit 4 Bitplanes (=1
          6 Farben) */
85 6s     HIRES,CUSTOMSCREEN,NULL,
86 Ix     (UBYTE *)"FrakTal-Berge V2.6 von Tobias Dierig & Chris Em
          merich",
87 uK     NULL,NULL,};
88 mp0 /* Die ganzen Texte für die Gadgets */
89 8w  struct IntuiText Texts1    =
90 lr1 {1,0,JAM2,173,1,NULL,(UBYTE *)string1,NULL};
91 E30 struct IntuiText Texts2    =
92 qx1 {1,0,JAM2,173,1,NULL,(UBYTE *)string2,NULL};
93 KAO struct IntuiText Texts3    =
94 v31 {1,0,JAM2,173,1,NULL,(UBYTE *)string3,NULL};
95 QHO struct IntuiText Texts4    =
96 091 {1,0,JAM2,173,1,NULL,(UBYTE *)string4,NULL};
97 SCO struct IntuiText TextWI    =
98 WL1 {1,1,JAM1,203,1,NULL,(UBYTE *)"Winkel",&Texts1};
99 9U0 struct IntuiText TextFA    =
100 UY1 {1,1,JAM1,203,1,NULL,(UBYTE *)"Faktor",&Texts2};
101 RQ0 struct IntuiText TextZU    =
102 Bu1 {1,1,JAM1,203,1,NULL,(UBYTE *)"Zufall",&Texts3};
103 oK0 struct IntuiText TextME    =
104 bQ1 {1,1,JAM1,203,1,NULL,(UBYTE *)"Meer",&Texts4};
105 Tb0 struct IntuiText TextST1   =
106 Ek1 {1,1,JAM1,5,41,NULL,(UBYTE *)"1",NULL};
107 aj0 struct IntuiText TextST2   =
108 Ip1 {1,1,JAM1,5,41,NULL,(UBYTE *)"2",NULL};
109 hr0 struct IntuiText TextST3   =
110 Mu1 {1,1,JAM1,5,41,NULL,(UBYTE *)"3",NULL};
111 oz0 struct IntuiText TextST4   =
112 Qz1 {1,1,JAM1,5,41,NULL,(UBYTE *)"4",NULL};
113 v70 struct IntuiText TextST5   =
114 U41 {1,1,JAM1,5,41,NULL,(UBYTE *)"5",NULL};
115 2F0 struct IntuiText TextST6   =
116 Y91 {1,1,JAM1,5,41,NULL,(UBYTE *)"6",NULL};
117 BV0 struct IntuiText TextR4    =
118 oR1 {1,1,JAM1,17,6,NULL,(UBYTE *)"4",NULL};
119 HcO struct IntuiText TextR5    =
120 sW1 {1,1,JAM1,17,6,NULL,(UBYTE *)"5",NULL};
121 NjO struct IntuiText TextR6    =
122 wb1 {1,1,JAM1,17,6,NULL,(UBYTE *)"6",NULL};
123 Tq0 struct IntuiText TextR7    =
124 Og1 {1,1,JAM1,17,6,NULL,(UBYTE *)"7",NULL};
125 Zx0 struct IntuiText TextR8    =
126 411 {1,1,JAM1,17,6,NULL,(UBYTE *)"8",NULL};
127 lu0 struct IntuiText TextQUIT  =
128 881 {1,1,JAM1,34,6,NULL,(UBYTE *)"QUIT",NULL};
129 Tx0 struct IntuiText TextRANDA =
130 FJ1 {1,0,JAM2,18,6,NULL,(UBYTE *)"RAND  AN",NULL};
131 X20 struct IntuiText TextRANDB =
132 dd1 {1,0,JAM2,18,6,NULL,(UBYTE *)"RAND AUS",NULL};
133 I10 struct IntuiText TextAN    =
134 3Z1 {0,1,JAM2,7,1,NULL,(UBYTE *)"AN",NULL};
135 tI0 struct IntuiText TextAUS   =
136 YK1 {0,1,JAM2,3,1,NULL,(UBYTE *)"AUS",NULL};
137 rt0 struct IntuiText TextSAVE =
138 Vr1 {0,1,JAM2,3,1,NULL,(UBYTE *)"SAVE",NULL};
139 H30 struct IntuiText TextCANC =
140 tn1 {0,1,JAM2,3,1,NULL,(UBYTE *)"CANCEL",NULL};
141 ka0 /*Nun kommen die Werte für die Umrandungen der Gadgets. In d
    en
142 z8  Feldern stehen jeweils X- und Y-Koordinaten von den einzelne
    n Punkten.*/
143 RM  USHORT Pairs1[] = {-1,-1,-1,18,40,18,40,-1,-1,-1};
144 4i  USHORT Pairs2[] = {-1,-1,-1,18,100,18,100,-1,-1,-1};
145 f1  USHORT Pairs3[] = {-1,0,-1,10,30,10,30,0};
146 ey  USHORT Pairs4[] = {-1,0,-1,10,38,10,38,0};
147 Yr  USHORT Pairs5[] = {-1,0,-1,10,55,10,55,0};
148 J3  USHORT Pairs6[] = {-1,-1,-1,10,427,10,427,-1,-1,-1};
149 DZ  struct Border Bord1 ={1,1,1,0,JAM1,5,(SHORT *)&Pairs1,NULL};
150 N1  struct Border Bord2 ={1,1,1,0,JAM1,5,(SHORT *)&Pairs2,NULL};
151 Sv  struct Border Bord3 ={0,0,0,1,JAM2,4,(SHORT *)&Pairs3,NULL};
152 c7  struct Border Bord4 ={0,0,0,1,JAM2,4,(SHORT *)&Pairs4,NULL};
153 mJ  struct Border Bord5 ={0,0,0,1,JAM2,4,(SHORT *)&Pairs5,NULL};
154 wW  struct Border Bord6 ={0,0,1,0,JAM2,5,(SHORT *)&Pairs6,NULL};
155 9g  /*StringInfo für das Gadget zum Eingeben des Dateinamens*/
156 6Y  struct StringInfo StringInfo1 ={
157 DP3    (UBYTE *)&Stringbuff,(UBYTE *)&Undobuff,0,60,0,0,0,0,0,0,
       0,0,0};
158 n80 /* Hier kommen die Gadgets */
159 Zg  struct Gadget GadCANC =
160 rD2    {NULL,431,0,54,10,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
161 4i     BOOLGADGET,(APTR)&Bord5,0,&TextCANC,0,NULL,CANC,0};
162 KS0 struct Gadget GadGETT =
163 uU2    {&GadCANC,2,2,426,10,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
164 jQ     STRGADGET,(APTR)&Bord6,0,NULL,0,(APTR)&StringInfo1,GETT,0}
       ;
165 XQ0 struct Gadget GadSAVE =
166 Ov2    {NULL,1,0,37,10,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
167 Mf     BOOLGADGET,(APTR)&Bord4,0,&TextSAVE,0,NULL,SAVE,0};
168 5b0 struct Gadget GadAUS =
169 V42    {&GadSAVE,71,0,30,10,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
170 VM     BOOLGADGET,(APTR)&Bord3,0,&TextAUS,0,NULL,AUS,0};
171 Ar0 struct Gadget GadAN =
172 7g2    {&GadAUS,40,0,30,10,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
173 gA     BOOLGADGET,(APTR)&Bord3,0,&TextAN,0,NULL,AN,0};
174 oB0 struct Gadget GadRAND =
175 GZ2    {NULL,340,55,100,20,GADGHNONE,GADGIMMEDIATE| RELVERIFY,
176 1J     BOOLGADGET,(APTR)&Bord2,0,&TextRANDA,0,NULL,BOU,0};
177 gN0 struct Gadget GadLI1 =
178 Iz2    {&GadRAND,470,5,160,70,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
179 J7     PROPGADGET,(APTR)&Imag162,0,NULL,0,(APTR)&Prop162,LI2,0};
180 pS0 struct Gadget GadLI2 =
181 BQ2    {&GadLI1,445,5,20,70,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
182 7x     PROPGADGET,(APTR)&Imag161,0,NULL,0,(APTR)&Prop161,LI1,0};
183 Xn0 struct Gadget GadQUIT =
184 DZ2    {&GadLI2,235,55,100,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
185 lr     BOOLGADGET,(APTR)&Bord2,0,&TextQUIT,0,NULL,QU,0};
186 jL0 struct Gadget GadR8 =
187 6D2    {&GadQUIT,190,55,40,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
188 m1     BOOLGADGET,(APTR)&Bord1,0,&TextR8,0,NULL,GR8,0};
189 hI0 struct Gadget GadR7 =
190 ce2    {&GadR8,145,55,40,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
191 gt     BOOLGADGET,(APTR)&Bord1,0,&TextR7,0,NULL,GR7,0};
192 fF0 struct Gadget GadR6 =
193 4G2    {&GadR7,100,55,40,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
194 al     BOOLGADGET,(APTR)&Bord1,0,&TextR6,0,NULL,GR6,0};
195 dC0 struct Gadget GadR5 =
196 eY2    {&GadR6,55,55,40,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
197 Ud     BOOLGADGET,(APTR)&Bord1,0,&TextR5,0,NULL,GR5,0};
198 b90 struct Gadget GadR4 =
199 qz2    {&GadR5,10,55,40,20,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
200 OV     BOOLGADGET,(APTR)&Bord1,0,&TextR4,0,NULL,GR4,0};
201 hR0 struct Gadget GadME =
202 RN2    {&GadR4,190,41,170,9,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
203 h3     PROPGADGET,(APTR)&Imag10,0,&TextME,0,(APTR)&Prop10,ME,0};
204 s50 struct Gadget GadZU =
205 ft2    {&GadME,190,29,170,9,GADGHCOMP,GADGIMMEDIATE| RELVERIFY,
```

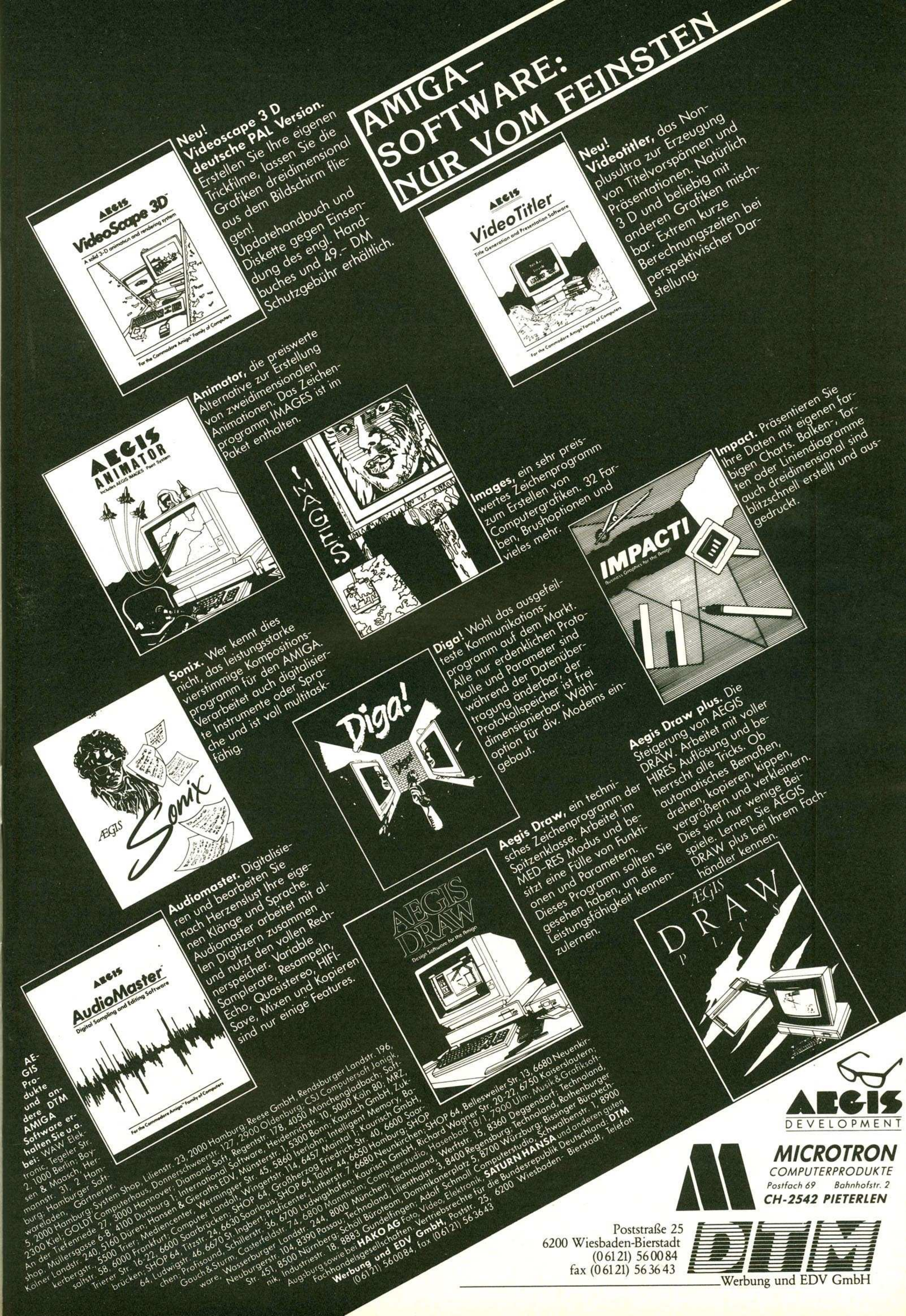
```
d9     PROPGADGET,(APTR)&Imag9,0,&TextZU,0,(APTR)&Prop9,ZU,0};
1a0  struct Gadget GadFA =
gK2    {&GadZU,190,17,170,9,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
iW     PROPGADGET,(APTR)&Imag8,0,&TextFA,0,(APTR)&Prop8,FA,0};
om0  struct Gadget GadWI =
GA2    {&GadFA,190,5,170,9,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
s5     PROPGADGET,(APTR)&Imag7,0,&TextWI,0,(APTR)&Prop7,WI,0};
5k0  struct Gadget GadST6 =
Qf2    {&GadWI,160,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
C3     PROPGADGET,(APTR)&Imag6,0,&TextST6,0,(APTR)&Prop6,ST6,0};
210  struct Gadget GadST5 =
2h2    {&GadST6,130,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
xk     PROPGADGET,(APTR)&Imag5,0,&TextST5,0,(APTR)&Prop5,ST5,0};
zm0  struct Gadget GadST4 =
vW2    {&GadST5,100,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
iR     PROPGADGET,(APTR)&Imag4,0,&TextST4,0,(APTR)&Prop4,ST4,0};
wn0  struct Gadget GadST3 =
BW2    {&GadST4,70,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
T8     PROPGADGET,(APTR)&Imag3,0,&TextST3,0,(APTR)&Prop3,ST3,0};
to0  struct Gadget GadST2 =
s02    {&GadST3,40,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
Ep     PROPGADGET,(APTR)&Imag2,0,&TextST2,0,(APTR)&Prop2,ST2,0};
qp0  struct Gadget GadST1 =
ZG2    {&GadST2,10,5,20,40,GADGHCOMP,GADGIMMEDIATE|RELVERIFY,
zW     PROPGADGET,(APTR)&Imag1,0,&TextST1,0,(APTR)&Prop1,ST1,0};
c00  /*3 eigene Fenster zur Darstellung und für die Gadgets*/
ak   struct NewWindow Window1 ={0,10,640,246,0,1,NULL,BORDERLESS,
iy3    NULL,NULL,NULL,NULL,NULL,0,0,0,0,CUSTOMSCREEN};
dm0  struct NewWindow Window2 ={0,175,640,81,0,1,GADGETDOWN|GADGE
     TUP,
S63    NULL,&GadST1,NULL,NULL,NULL,NULL,0,0,0,0,CUSTOMSCREEN};
sr0  struct NewWindow Window3 ={486,0,102,10,1,1,GADGETUP,BORDERL
     ESS,
tc3    &GadAN,NULL,NULL,NULL,NULL,0,0,0,0,CUSTOMSCREEN};
5F0  struct NewWindow Window4 ={0,0,485,13,1,1,GADGETUP,BORDERLES
     S,
2b3    &GadGETT,NULL,NULL,NULL,NULL,0,0,0,0,CUSTOMSCREEN};
jN0  /*Routine zeichnet Dreiecke (evt. als Wasser oder mit Umrand
     ung*/
Xg   VOID Drawit()
qJ3    {
Oy     int i,ff;
s0     float va;
tR     AreaMove (rp,(long)(dec[2][0]),(long)(dec[2][1]));
ol     for (i=0;i<=2;i++)     /* Dreieck malen */
Ad6      AreaDraw (rp,(long)(dec[i][0]),(long)(dec[i][1]));
MV3    f = (f < 03) ? 03 : f;    /* Farbe in der Begrenzung */
bm     f = (f > 15) ? 15 : f;
4o     if (dec[2][2] <= meer && dec[1][2] <= meer && dec[0][2]
        <= meer)
5v6      f = 2;            /* unter Meeresspiegel, dann blau */
WJ3    if (bound == 0)
dv6      ff = f;
vi3    else
x06      ff = 1;
P53    SetAPen(rp,(long)(f*inh));
ak     SetOPen(rp,(long)(ff));
ln     AreaEnd(rp);   /* Dreieck ausmalen */
Bg     }
E00  /*Routine berechnet Normale auf Dreieck und damit die Farbe*
     /
0F   VOID threedee()
Ad3    {
Yv     int la;
a9     float a[3],b[3],n[3];      /* Vektoren */
Zk     float bl,skal,c;
H6     for (la=0;la<=2;la++)
Fi6      {
xj       a[la] = dec[0][la]-dec[1][la];   /* Vektoren berechnen
         */
ll       b[la] = dec[2][la]-dec[1][la];
Mr       }
6d3    n[0] = a[1]*b[2]-a[2]*b[1];     /* X Normale auf Dreieck *
       /
Rz     n[1] = a[2]*b[0]-a[0]*b[2];     /* Z */
Ez     n[2] = a[0]*b[1]-a[1]*b[0];     /* Y */
Cn     bl   = sqrt(n[0]*n[0]+n[1]*n[1]+n[2]*n[2]);
uW     skal = l1*n[0]+l2*n[1]+l3*n[2];
7I     c = (bl*l != 0) ? skal/(bl*l) : 0;   /* Farbwert ermittel
       n */
6s     f = ((c+1)*7)+2;          /* Farbe bestimmen */
NC     for(la=0;la<=2;la++)
Ru6      {
Vo         if (dec[la][2] <= meer)     /* Höhe unter Meer */
5U9          dec[la][2] = meer;      /* Meeresspiegel sichtbar */

wq6        dec[la][1] = (basis-dec[la][1]*si-dec[la][2]*co);
Z4         }
a53      }
U60  /*Routine zur Berechnung und Ausgabe eines Dreiecks*/
T5   VOID Buildit()
Z23    {
9K     int x,y;
ov     for (y = 0;y < maxst;y++)      /* Alle Koordinaten durchge
       hen */
c56      {
1C       for (x = 0;x < (maxst-y);x++)
e79        {
Fd         dec[0][0] = ((x-(maxst-(y+1))/2)*sx+xmitte);  /* Ko
           ordinaten */
db         dec[1][0] = ((x-(maxst-y)/2)*sx+xmitte);       /* er
           mitteln  */
3G         dec[2][0] = (((x+1)-(maxst-y)/2)*sx+xmitte);
dU         dec[0][1] = ((maxst-(y+1))*sy);
2N         dec[1][1] = ((maxst-y)*sy);
9Q         dec[2][1] = ((maxst-y)*sy);
jF         dec[0][2] = kor[x][y+1];
pp         dec[1][2] = kor[x][y];
fJ         dec[2][2] = kor[x+1][y];
69         threedee();   /* Zeichenkoordinaten berechnen */
3s         Drawit();     /* Dreieck drucken */
RV         if (x == 0)   /* Am Rand nach unten löschen */
rKC          {
ro           SetAPen(rp,0L);
0B           SetOPen(rp,0L);
M6           AreaMove(rp,(long)(dec[0][0]),(long)(dec[0][1]+1
             ));
sE           AreaDraw(rp,(long)(dec[1][0]),(long)(dec[1][1]+1
             ));
id           AreaDraw(rp,(long)(dec[1][0]),246L);
eY           AreaDraw(rp,(long)(dec[0][0]),246L);
cT           AreaEnd(rp);
3Y           }
mJ9        if (x != (maxst-y-1))    /* gegenüberliegendes Dreie
           ck berechnen */
1UC          {
Qe           dec[2][0] = ((x-(maxst-(y+1))/2)*sx+xmitte);
GL           dec[1][0] = (((x+1)-(maxst-(y+1))/2)*sx+xmitte);
EZ           dec[0][0] = (((x+1)-(maxst-y)/2)*sx+xmitte);
Cv           dec[2][1] = ((maxst-(y+1))*sy);
7u           dec[1][1] = ((maxst-(y+1))*sy);
Kj           dec[0][1] = ((maxst-y)*sy);
ZA           dec[2][2] = (kor[x][y+1]);
TK           dec[1][2] = (kor[x+1][y+1]);
NU           dec[0][2] = (kor[x+1][y]);
L9           threedee();
WE           Drawit();
Hm           }
7u9        else
FiC          {
FC           SetAPen(rp,0L);
OZ           SetOPen(rp,0L);
kU           AreaMove(rp,(long)(dec[0][0]),(long)(dec[0][1]+1
             ));
Om           AreaDraw(rp,(long)(dec[2][0]),(long)(dec[2][1]+1
             ));
B7           AreaDraw(rp,(long)(dec[2][0]),246L);
2w           AreaDraw(rp,(long)(dec[0][0]),246L);
Or           AreaEnd(rp);
Rw           }
Sx9        }
Ty6      }
Uz3    }
w80  /*Exponentfunktion*/
Cr   int expon(basis,exponent)
JC3    int basis,exponent;
Ux     {
3Z     int i;
54     int erg = 1;
n0     for (i = 1;i <= exponent;i++)
w36      erg *= basis;
```

Listing. Norwegische Fjorde auf Knopfdruck in fünf Minuten mit »Fraktalberge« (Fortsetzung)

```
   return erg;
   }
/*Routine legt Farbwerte für Ausgabe fest*/
VOID setcolor()
   {
   long ff;
   SetRGB4(&MyScreen->ViewPort,0L,0L,0L,0L); /* Farbe schwa
   rz */
   SetRGB4(&MyScreen->ViewPort,2L,0L,0L,13L);/* Farbe blau
   */
   for (ff=3;ff<=15;ff++)    /* Grautöne erzeugen */
      SetRGB4(&MyScreen->ViewPort,ff,ff,ff,ff);
   }
/*Routine zum Erstellen der Landschaft*/
calc(x,y,r,stufen)
   int x,y,r,stufen;
   {
   int z1,z2,rr1,rr2;
   if (stufen >= 1) /* maximale Tiefe nich erreicht */
      {
      z1 = kor[x][y];            /* Das Dreieck wird in vier w
      eitere */
      z2 = kor[x+stufen*r][y]; /* zerstückelt und die Routin
      e wird */
      rr1 = stufen*r;            /* aufgerufen */
      rr2 = stufen/2*r;
      if (kor[x+rr2][y] == 0)  /* Die Höhen werden zuerst */
         kor[x+rr2][y] = mittel(z1,z2,stufen); /* ermittelt
         */
      z2=kor[x][y+rr1];
      if (kor[x][y+rr2] == 0)
         kor[x][y+rr2] = mittel(z1,z2,stufen);
      z1=kor[x+rr1][y];
      if (kor[x+rr2][y+rr2] == 0 )
         kor[x+rr2][y+rr2] = mittel(z1,z2,stufen);
      calc(x,y,r,stufen/2);     /* Aufteilung in weitere Drei
      ecke */
      calc(x+stufen/2*r,y,r,stufen/2);
      calc(x,y+stufen/2*r,r,stufen/2);
      calc(x+stufen/2*r,y+stufen/2*r,-r,stufen/2);
      }
   }
/*Routine zum Berechnen des Mittelwertes*/
int mittel(m1,m2,fakt)
   int m1,m2;
   int fakt;
   {
   int m;
   m = (m1+m2)/2;
   m += (rand(1)/16348-.5)*vakt*fakt/maxst;
   return(m);
   }
/*Routine zum Löschen des Bildschirms*/
VOID clsscr()
   {
   SetAPen(rp,0L);
   SetOPen(rp,0L);
   RectFill(rp,0L,0L,640L,246L); /* schwarz ausmalen */
   }
/*Routine zur Umwandlung einer INT-Zahl in einen String*/
itoa (n,s)
int n;
char s[];
   {
   if (n >= 99 || n <= -99) n=99; /* Innerhalb Begrenzung
   */
   if (n <= -1)
      {
      n = -n;
      s[0] = 45;   /* negatives Vorzeichen */
      }
   else
      {
      if (n >= 1) s[0] = 43;  /* positives Vorzeichen */
      if (n == 0) s[0] = 177; /*  für null */
      }
   s[2] = (n % 10)+48;   /* Einerstelle */
   s[1] = ((n/10) % 10)+48;  /* Zehnerstelle */
   s[3] = 0;
   }
/*Routine zur Darstellung des Ansichtsdreieck zum Einstellen
 der Werte*/
VOID Line_it(fll)
int fll;
   {
   int ii,jj;
   clsscr();   /* Bildschirm löschen */
   grad  = grad1;
   maxst = (float)(expon(2,(grad-1)));
   s   = s1;
   inh = 1;
   if (fll == 1)
      {
      maxst = 2;
      inh   = 0;
      }
   else
      {
      WindowToBack(MyWindow1);
      }
   for (ii=0;ii<=maxst;ii++)
      {
      for (jj=0;jj<=maxst;jj++)
         kor[ii][jj]=0;
      }
   kor[0][0]                                = st[1];    /* Stüt
   zpunkte */
   kor[(long)(maxst/2)][0]                  = st[2];
   kor[(long)(maxst)][0]                    = st[3];
   kor[0][(long)(maxst/2)]                  = st[4];
   kor[(long)(maxst/2)][(long)(maxst/2)] = st[5];
   kor[0][(long)(maxst)]                    = st[6];
   meer = meer1;
   vakt = vakt1;
   srand(s);
   sx = (int)(540/maxst);
   sy = (int)(sx*.33);
   co = cos(winkel*k);
   si = sin(winkel*k);
   if (fll == 0)
     calc(0,0,1,(int)(maxst));   /* Berechnung starten */
   Buildit();
   }
/*Routine zum Abspeichern des Bildschirms im IFF-Format*/
Speich()
   {
   SaveWindow = OpenWindow(&Window4); /* Fenster öffnen */
   RectFill(SaveWindow->RPort,431L,0L,485L,10L);
   RefreshGList(&GadCANC,SaveWindow,NULL,1L);
   class = GADGETDOWN;
   do           /* Gadgets abfragen */
      {
      if ((Message = (struct IntuiMessage *)GetMsg(SaveWindo
      w->UserPort)) != 0)
         {
         class = Message->Class;
         code = Message->Code;
         Gadg = ((struct Gadget *)(Message->IAddress))->Ga
         dgetID;
         ReplyMsg(Message);
         if ((class == GADGETUP) && (Gadg == GETT)) SaveFrak
         t();
         }
      }
   while (class != GADGETUP);
   CloseWindow(SaveWindow);  /* Fenster wieder zu */
   }
SaveFrakt()
   {
   int i;
   for (i=0;i<15;i++)  /* Farbpalette vorbereiten */
      {
      ColorMap1 = GetRGB4((((struct ViewPort *)(&(MyScreen->
      ViewPort)))
                             ->ColorMap),(long)i);
      ColorMap2[3*i+0] = (UBYTE)(((ColorMap1 >> 8) & 15) <
      < 4);
      ColorMap2[3*i+1] = (UBYTE)(((ColorMap1 >> 4) & 15) <
      < 4);
      ColorMap2[3*i+2] = (UBYTE)(( ColorMap1      & 15) <<
       4);
```

Listing. Norwegische Fjorde auf Knopfdruck in fünf Minuten mit »Fraktalberge« (Fortsetzung)


AMIGA-SOFTWARE: NUR VOM FEINSTEN
Neu! Videoscape 3 D deutsche PAL Version. Erstellen Sie Ihre eigenen Trickfilme, lassen Sie die Grafiken dreidimensional aus dem Bildschirm fliegen!
Updatehandbuch und Diskette gegen Einsendung des engl. Handbuches und 49.- DM Schutzgebühr erhältlich.
AEGIS VideoScape 3D
A solid 3-D animation and rendering system
For the Commodore Amiga Family of Computers
Neu! Videotitler, das Nonplusultra zur Erzeugung von Titelvorspännen und Präsentationen. Natürlich 3 D und beliebig mit anderen Grafiken mischbar. Extrem kurze Berechnungszeiten bei perspektivischer Darstellung.
AEGIS VideoTitler
Title Generation and Presentation Software
For the Commodore Amiga Family of Computers
Animator, die preiswerte Alternative zur Erstellung von zweidimensionalen Animationen. Das Zeichenprogramm IMAGES ist im Paket enthalten.
AEGIS ANIMATOR
Includes AEGIS IMAGES Paint System
IMAGES
AEGIS
Images, ein sehr preiswertes Zeichenprogramm zum Erstellen von Computergrafiken. 32 Farben, Brushoptionen und vieles mehr.
Impact. Präsentieren Sie Ihre Daten mit eigenen farbigen Charts. Balken-, Torten oder Liniendiagramme auch dreidimensional sind blitzschnell erstellt und ausgedruckt.
IMPACT!
Business Graphics for the Amiga
Sonix. Wer kennt dies nicht, das leistungsstarke vierstimmige Kompositionsprogramm für den AMIGA. Verarbeitet auch digitalisierte Instrumente oder Sprache und ist voll multitaskfähig.
AEGIS Sonix
Diga! Wohl das ausgefeilteste Kommunikationsprogramm auf dem Markt. Alle nur erdenklichen Protokolle und Parameter sind während der Datenübertragung änderbar, der Protokollspeicher ist frei dimensionierbar. Wähloption für div. Modems eingebaut.
Diga!
Aegis Draw plus. Die Steigerung von AEGIS DRAW. Arbeitet mit voller HIRES Auflösung und beherrscht alle Tricks. Ob automatisches Bemaßen, drehen, kopieren, kippen, vergrößern und verkleinern. Dies sind nur wenige Beispiele. Lernen Sie AEGIS DRAW plus bei Ihrem Fachhändler kennen.
Aegis Draw, ein technisches Zeichenprogramm der Spitzenklasse. Arbeitet im MED-RES Modus und besitzt eine Fülle von Funktionen und Parametern. Dieses Programm sollten Sie gesehen haben, um die Leistungsfähigkeit kennenzulernen.
AEGIS DRAW PLUS
Audiomaster. Digitalisieren und bearbeiten Sie nach Herzenslust Ihre eigenen Klänge und Sprache. Audiomaster arbeitet mit allen Digitizern zusammen und nutzt den vollen Rechnerspeicher. Variable Samplerate, Resampeln, Echo, Quasistereo, HIFI-Save, Mixen und Kopieren sind nur einige Features.
AEGIS AudioMaster
Digital Sampling and Editing Software
For the Commodore Amiga Family of Computers
AEGIS DRAW
Design Software for the Amiga
AEGIS Produkte und andere DTM AMIGA Software erhalten Sie u.a. bei: WAW Elektronic, Tegeler Str. 2, 1000 Berlin; Boysen & Maasch, Hermannstr. 31, 2 Hamburg; Hamburger Softwareladen, Gärtnerstr. 5, 2000 Hamburg; System Shop, Lilienstr. 23, 2000 Hamburg; Reese GmbH, Rendsburger Landstr. 196, 2300 Kiel; GOLDT Computerhaus, Donnerschweerstr. 127, 2900 Oldenburg; CSJ Computersoft Jonigk, An der Tiefenriede 27, 3000 Hannover; Diamond Soft, Regentstr. 178, 4050 Mönchengladbach; Softshop, Müllersgasse 6-8, 4100 Duisburg 1; International Software, Heidenrichstr. 10, 5000 Köln 80; MRZ, Kolner Landstr. 240, 5160 Düren; Hansen & Gieraths EDV, Münsterstr. 1, 5300 Bonn; Kaurisch GmbH, Zukerbergstr., 5500 Trier; Mediencenter, Wermingser Str. 45, 5860 Iserlohn; Intelligent Memory, Basaltstr. 58, 6000 Frankfurt; Computer Landolt, Wingertstr. 114, 6457 Maintal 1; Kaurisch GmbH, Trierer Str. 16-22, 6600 Saarbrücken; SHOP 64, Großherzog Friedrich-Str. 40, 6600 Saarbrücken; SHOP 64, Titzstr. 24, 6630 Saarlouis; SHOP 64, Talstr. 44, 6650 Homburg; SHOP 64, Ludwigstr. 46, 6670 St. Ingbert; Proficenter, Lutherstr. 7, 6680 Neunkirchen; SHOP 64, Bellesweiler Str. 13, 6680 Neuenkirchen; Profisound, Schillerstr. 36, 6700 Ludwigshafen; Kaurisch GmbH, Richard Wagner Str. 20-22, 6750 Kaiserslautern; Gauch & Sturm, Casterfeldstr. 74, 6800 Mannheim; Computerstudio, Hasenbad 18/1, 7900 Ulm; Musik & Grafiksoftware, Wasserburger Landstr. 244, 8000 München; Technoland, Werftstr. 15, 8360 Deggendorf; Technoland, Neuburgerstr. 104, 8390 Passau; Technoland, Lilienthalstr. 3, 8400 Regensburg; Technoland, Rothenburger Str. 451, 8500 Nürnberg; Schöll Büroteam, Dominikanerplatz 5, 8700 Würzburg; Bissinger Bürotechnik, Industriestr. 18, 8883 Gundelfingen; Adolf Schmoll Computerstudio, Schwalbenstr. 1, 8900 Augsburg sowie bei HAKO AG Foto, Video, Elektronik, SATURN HANSA und anderen guten Fachhandelsgesellschaften. Vertriebsrechte für die Bundesrepublik Deutschland: DTM Werbung und EDV GmbH, Poststr. 25, 6200 Wiesbaden- Bierstadt, Telefon (06121) 560084, fax (06121) 563643
AEGIS DEVELOPMENT
MICROTRON COMPUTERPRODUKTE
Postfach 69 Bahnhofstr. 2
CH-2542 PIETERLEN
Poststraße 25
6200 Wiesbaden-Bierstadt
(0 61 21) 56 00 84
fax (0 61 21) 56 36 43
DTM
Werbung und EDV GmbH

```
yT        }
gm3       /* Routinenaufruf für Routine aus jiff_save.c von Aztek C Disk */
yF        write_iff(Stringbuff,ColorMap2,MyWindow->RPort->BitMap,
BxD                 (SHORT)0,(SHORT)0,(UWORD)640,(SHORT)1);
2X3       }
HNO /* -Hauptprogramm- */
17  main()
1U3       {
QK        int i,j;
1M        strcpy(string1,"+15");string1[3]=0;    /* Strings für Gadgets */
GG        strcpy(string2,"+50");string2[3]=0;
ei        strcpy(string3,"+25");string3[3]=0;
af        strcpy(string4,"00");string4[3]=0;
Xh        winkel=15;
Xw        vakt1=50;
wO        s1=25;
D4        meer1=0;
nP        maxl = MAX(abs(l1),MAX(abs(l2),abs(l3)));
zd        l1 /= maxl;    /* Lichtvektor vorbereiten zum späteren Rechnen */
rz        l2 /= maxl;
v4        l3 /= maxl;
4U        l = sqrt(l1*l1+l2*l2+l3*l3); /* Länge des Vektors */
FM        if((IntuitionBase = (struct IntuitionBase *)
CXL                       OpenLibrary("intuition.library",0L)) ==
                          NULL )
I16          {
tx           printf("Intuition nicht da !!!\n");
5V           exit(FALSE);
Pu           }
d03       if((GfxBase = (struct GfxBase *)
JXF                  OpenLibrary("graphics.library",0L)) == NULL )
Or6          {
b8           printf("Graphics nicht da !!!\n");
1K           CloseLibrary(IntuitionBase);
Cc           exit(FALSE);
W1           }
QD3       Prop1.Flags      = Prop2.Flags = Prop3.Flags = Prop4.Flags =
1iM                          Prop5.Flags = Prop6.Flags = Prop161.Flags =
90                           AUTOKNOB| FREEVERT;
Pc3       Prop7.Flags      = Prop8.Flags = Prop9.Flags = Prop10.Flags =
fPM                          AUTOKNOB| FREEHORIZ;
ni3       Prop162.Flags    = AUTOKNOB| FREEHORIZ| FREEVERT;
ir        Prop1.VertPot    = Prop2.VertPot = Prop3.VertPot = Prop4.VertPot =
WiM                          Prop5.VertPot = Prop6.VertPot = MAXPOT/2;
fI3       Prop7.HorizPot   = 13107;
hK        Prop8.HorizPot   = 32768;
M1        Prop9.HorizPot   = 16384;
B1        Prop10.HorizPot  = 32768;
RD        Prop161.VertPot  = 65535;
nF        Prop162.HorizPot = 38768   ; Prop162.HorizBody = MAXBODY/15;
J1        Prop162.VertPot  = MAXPOT/2; Prop162.VertBody  = MAXBODY/15;
fX        Prop1.VertBody   = Prop2.VertBody = Prop3.VertBody = Prop4.VertBody =
2QM                          Prop5.VertBody = Prop6.VertBody = Prop161.VertBody =
Wd                           MAXBODY/20;
Oc3       Prop7.HorizBody  = Prop8.HorizBody = Prop9.HorizBody =
epM                          Prop10.HorizBody = MAXBODY/50;
Gs0 /*Screen und Windows öffnen*/
oB3       MyScreen        = OpenScreen(&TheScreen);
Co        Window1.Screen = MyScreen;        /* Alle Fenster in unserem Screen */
ni        Window2.Screen = MyScreen;
rn        Window3.Screen = MyScreen;
vs        Window4.Screen = MyScreen;
4t        MyWindow        = OpenWindow(&Window1);   /* Fenster auf */
id        MyWindow1       = OpenWindow(&Window2);
bD        IconWindow      = OpenWindow(&Window3);
Pw        rp = MyWindow->RPort;           /* der RastPort */
5b        SetAPen(IconWindow->RPort,1L);
82        RectFill(IconWindow->RPort,0L,0L,101L,10L);
yp        RefreshGadgets(&GadAN,IconWindow,NULL);
Ou        /* Jetzt wird Speicher für die Area-Befehle reserviert */
Jp        InitArea (&MyAreaInfo,AreaBuffer,5L);
A6        rp->AreaInfo = &MyAreaInfo;
Pu        Pointer = AllocRaster(640L,256L);
gI        if (Pointer == 0 )
5Y6          {
dV           printf("No free Space for Areafill !!!\n");
Y1           CloseWindow(MyWindow1);
Rd           CloseWindow(MyWindow);
71           CloseWindow(IconWindow);
8P           CloseScreen(MyScreen);
fB           CloseLibrary(GfxBase);
X6           CloseLibrary(IntuitionBase);
G1           exit(TRUE);
In           }
Zn3       InitTmpRas (&MyTmpRas,(PLANEPTR)Pointer,RASSIZE(640,256));
jm        rp->TmpRas = &MyTmpRas;
mN        setcolor();
BeO /*Abfrage, ob Gadget betätigt wurde*/
KF3       lo:
2K        if ((Message = (struct IntuiMessage *)GetMsg(IconWindow->UserPort)) != 0)
Lo6          {
vu           class = Message->Class; /* Art der Nachricht */
pY           code = Message->Code;
aB           Gadg = ((struct Gadget *)(Message->IAddress))->GadgetID;
KuE                   /* Betätigtes Gadget */
qg6          ReplyMsg(Message);    /* Erstmal antworten */
1o           if (class == GADGETUP)  /* Gadget gewählt */
Sv9             {
Az              switch(Gadg)
UxC                {
12                 case AN:
WzF                   {
XX                    WindowToFront(MyWindow1);
XW                    goto lo;
d8                    }
t6C                case AUS:
b4F                   {
OB                    WindowToBack(MyWindow1);
cb                    goto lo;
iD                    }
BuC                case SAVE:
g9F                   {
ZW                    Speich();
mH                    }
nIC                }
oJ9             }
pK6          }
A03       /* Folgende Gadgets werden nur nach AN abgefragt */
M1        /* da vorher das Fenster im Hintergrund liegt */
qk        if ((Message = (struct IntuiMessage *)GetMsg(MyWindow1->UserPort)) != 0)
pI6          {
eY           class = Message->Class;
J2           code = Message->Code;
4f           Gadg = ((struct Gadget *)(Message->IAddress))->GadgetID;
y3           ReplyMsg(Message);
uY           if (class == GADGETUP)
v09             {
dS              switch(Gadg)
xQC                {
5j                 case QU:
WPF                   goto ende;
pkC                case GR4:     /* Tiefe der Berechnung */
1UF                   {
Js                    grad1 = 4;
4Z                    Line_it(0);  /* Landschaft zeichnen */
32                    goto lo;
9e                    }
WIC                case GR5:
7aF                   {
U4                    grad1 = 5;
pn                    Line_it(0);
98                    goto lo;
Fk                    }
gTC                case GR6:
DgF                   {
fG                    grad1 = 6;
```

```
vt                  Line_it(0);
FE                  goto lo;
Lq                  }
qeC             case GR7:
JmF                 {
qS                  grad1 = 7;
1z                  Line_it(0);
LK                  goto lo;
Rw                  }
OpC             case GR8:
PsF                 {
1e                  grad1 = 8;
75                  Line_it(0);
RQ                  goto lo;
X2                  }
OHC             case ST1: /* Stützpunkt wird verändert */
VyF                 {
1q                  st[1]=(int)(-((long)(Prop1.VertPot)-32768)/10
                    00L);
IH                  Line_it(1);
XW                  goto lo;
d8                  }
IFC             case ST2:
b4F                 {
v5                  st[2]=(int)(-((long)(Prop2.VertPot)-32768)/10
                    00L);
ON                  Line_it(1);
dc                  goto lo;
jE                  }
SQC             case ST3:
hAF                 {
8K                  st[3]=(int)(-((long)(Prop3.VertPot)-32768)/10
                    00L);
UT                  Line_it(1);
ji                  goto lo;
pK                  }
cbC             case ST4:
nGF                 {
LZ                  st[4]=(int)(-((long)(Prop4.VertPot)-32768)/10
                    00L);
aZ                  Line_it(1);
po                  goto lo;
vQ                  }
mmC             case ST5:
tMF                 {
Yo                  st[5]=(int)(-((long)(Prop5.VertPot)-32768)/10
                    00L);
gf                  Line_it(1);
vu                  goto lo;
1W                  }
wxC             case ST6:
zSF                 {
13                  st[6]=(int)(-((long)(Prop6.VertPot)-32768)/10
                    00L);
ml                  Line_it(1);
10                  goto lo;
7c                  }
TGC             case ZU:
5YF                 {
HK                  s1=(int)(((long)(Prop9.HorizPot)-32768)/656L+
                    49);
HJ                  itoa(s1,string3);  /* Neuen Wert des Gadgets
                    darstellen */
yJ                  RefreshGList(&GadZU,MyWindow1,NULL,1L); /*Gad
                    get erneuern*/
87                  goto lo;
EJ                  }
OiC             case ME:
CfF                 {
gI                  meer1=(int)((((long)(Prop10.HorizPot))-32768)
                    /1093L);
R6                  itoa(meer1,string4);
Ff                  RefreshGList(&GadME,MyWindow1,NULL,1L);
FE                  goto lo;
Lq                  }
5EC             case FA:
JmF                 {
mR                  vakt1=(int)(((long)(Prop8.HorizPot)-32768)/65
                    6L+49);
4u                  itoa(vakt1,string2);
ap                  RefreshGList(&GadFA,MyWindow1,NULL,1L);
ML                  goto lo;
Sx                  }
8gC             case WI:
QtF                 {
NY                  winkel=(float)((long)(Prop7.HorizPot)/937L);
OW                  itoa((int)(winkel),string1);
R5                  RefreshGList(&GadWI,MyWindow1,NULL,1L);
FE                  Line_it(1);
UT                  goto lo;
a5                  }
z3C             case BOU:  /* Jeder Druck wechselt Zustand von A
                 und B */
Y1F                 {
gr                  if (bound==0)
a3I                     {
TE                      bound=1;
Ri                      GadRAND.GadgetText = &TextRANDB;
hC                      }
XKF                 else
f8I                     {
VF                      bound=0;
Uk                      GadRAND.GadgetText = &TextRANDA;
mH                      }
Y4F                 RefreshGList(&GadRAND,MyWindow1,NULL,1L);
oJ                  }
b1C             case LI1: /* Der Lichtvektor wird verändert */
mFF                 {
Pm                  l1=(float)( ((float)(Prop162.HorizPot)-32768)
                     /6000);
F2                  l2=(float)(-(((float)(Prop162.VertPot)-32768)
                    )/6000);
d7                  l3=(float)((65535-(float)(Prop161.VertPot))/1
                    2000);
S4                  maxl = MAX(abs(l1),MAX(abs(l2),abs(l3)));
om                  if (maxl != 0)
sLI                     {
Ub                      l1 /= maxl;
Yg                      l2 /= maxl;
cl                      l3 /= maxl;
OV                      }
nAF                 l = sqrt(l1*l1+l2*l2+l3*l3);
2X                  }
wbC             case LI2:
OTF                 {
d0                  l1=(float)( ((float)(Prop162.HorizPot)-32768)
                     /6000);
TG                  l2=(float)(-(((float)(Prop162.VertPot)-32768)
                    )/6000);
rL                  l3=(float)((65535-(float)(Prop161.VertPot))/1
                    2000);
gI                  maxl = MAX(abs(l1),MAX(abs(l2),abs(l3)));
20                  if (maxl != 0)
nJI                     {           /* Neue Berechnung des Vektor
                        s */
ip                      l1 /= maxl;
mu                      l2 /= maxl;
qz                      l3 /= maxl;
Ej                      }
njF                 l=sqrt(l1*l1+l2*l2+l3*l3);
G1                  }
HmC             }
In9         }
Jo6     }
FE3     goto lo;
FGO /* Alles wieder schließen und ENDE */
pj3     ende:
rt      FreeRaster((PLANEPTR)Pointer,640L,256L); /* Speicher wied
        erherstellen */
1E      CloseWindow(MyWindow1);
eq      CloseWindow(MyWindow);
KE      CloseWindow(IconWindow);
Lc      CloseScreen(MyScreen);
s0      CloseLibrary(GfxBase);
kJ      CloseLibrary(IntuitionBase);
rP      exit(TRUE); /* Bye Bye */
V0      }
(C) 1988 M&T
```

Listing. Norwegische Fjorde auf Knopfdruck in fünf Minuten mit »Fraktalberge« (Schluß)

Schneller Überblick

Wenn Sie der Meinung sind, Ordnung sei das halbe Leben, dann ist »Etikett« wohl genau das Richtige für Sie. Dieses Programm druckt das Directory Ihrer Disketten auf genormte Endlosetiketten.

Die Etiketten sind in jedem besseren Schreibwaren- oder Büromaterialiengeschäft zu haben. Sie sind zum Beispiel von der Firma HERMA mit der Nummer 8008 erhältlich. Diese Größe entspricht den Original-Diskettenaufklebern, die meist den Diskettenpackungen schon beigefügt sind. Allerdings passen auf die Aufkleber nur vierzig Inhaltsverzeichniseinträge, wenn Sie den Epsondruckertreiber ausgewählt haben. Mit dem NEC P6 können Sie nur 36 Einträge drucken.

Der Aufbau der Druckmaske läßt sich mit dem DIR-Befehl im CLI vergleichen. Das Programm druckt auch alle Subdirectories zuerst und dann die übrigen Einträge aus dem Hauptbereich der Diskette. In der Kopf- und Fußzeile dienen Striche als Umrahmung, um das Inhaltsverzeichnis klar vom Diskettennamen, der in der obersten Zeile steht, hervorzuheben. Um möglichst viele Einträge zu drucken, wird der Subscript-Zeichensatz benutzt.

Das Programm wird ausschließlich mit der Maus gesteuert. Es wurden konsequent Requester benutzt, um mit dem Programm zu kommunizieren. Als erstes kommt die Frage nach dem Druckertyp, Epson oder NEC P6. Mit nicht ganz hundertprozentig kompatiblen Druckern funktioniert das Programm aber auch. Danach wird man aufgefordert, den Drucker einzuschalten. Ist dies geschehen, wird das »OK« mit der Maus übergeben. Nach der Beantwortung der Frage, ob eine Diskette eingelegt ist, beginnt das Programm im Laufwerk DF0: das Inhaltsverzeichnis einzulesen. Nach jedem Eintrag, den das Programm von der Diskette holt, muß mit »JA« oder »NEIN« geantwortet werden, ob man den Eintrag auf dem Etikett haben möchte oder nicht. Dies geschieht, indem man im Menübalken den JA- oder NEIN-Schalter anklickt. Diese Art der Datenübergabe wird bis jetzt selten genutzt, hat aber Vorteile, da kein Platz im Fenster benötigt wird. Neben dem Eintrag des Inhaltsverzeichnisses steht eine Zahl. Diese gibt an, wie viele Einträge noch auf dem Etikett frei sind. Ist alles ordnungsgemäß selektiert worden, fängt das Programm zu drucken an. Letztendlich wird dann noch gefragt, ob man eine weitere Diskette bearbeiten oder lieber das Programm beenden möchte. Ein kleiner Trick: Eine Rückkehr zum CLI oder zur Workbench ist auch möglich, wenn man CTRL-C drückt und noch eine Auswahl tätigt. Dies ist zum Beispiel notwendig, wenn eine Fehleingabe erfolgte und es deshalb nicht mehr sinnvoll ist, das Programm ordentlich abzuschließen.

```
WB RB
---------------------------------
*c                 *System
*l                 *devs
*s                 *T
*fonts             *libs
beweb              popcli
popcolours         BootGirl.data
CLI                ddsir
Calculator         telefon
FastMemFirst       CLImate
Disk.info          System.info
NoFastMem.info     SS
NoFastMem          cs
dcopy
---------------------------------
```

So sieht ein fertiger Ausdruck mit dem NEC P6 aus

Programmname:	Etikett
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4 oder Lattice-C V3.1
Aufrufe:	siehe Text

Wenn Sie den Aztec-C-Compiler besitzen, übersetzen Sie bitte das Programm mit dem Aufruf

```
cc -s Etikett.c
```

Linken Sie danach den Objectcode mit:

```
ln Etikett.o -lc.
```

Hier die Aufrufe bei der Verwendung des Lattice-C-Compilers:

```
lc etikett
blink lib:c.o,etikett.o lib lib:lc.lib,lib:amiga.lib
```

Ist das Programm compiliert und gelinkt, dann können Sie es vom CLI aus starten. Natürlich ist auch ein Start von der Workbench aus möglich. Erstellen Sie dazu das File Etikett.info mit dem Programm IconEd Ihrer Workbenchdiskette. Um innerhalb der Menüleiste eine Abfrage zu realisieren, hat der Autor einfach den Abfragetext (*JA* bzw. *NEIN*) dem Menüleistentext übergeben. Es müssen dann nur noch die Mauskoordinaten abgefragt werden. Schreiben Sie deshalb den Programmteil »Window öffnen und Meldung ausgeben« genau ab und vergessen Sie kein Leerzeichen. Möchten Sie trotzdem die Menüleiste anders gestalten, so können Sie die Koordinaten bei der Mausabfrage ändern. Dies sollte bei der Anweisung

```
if (x>=383 && x<=413 && y>=0 && y<=10)
```

für JA geschehen. Bei NEIN machen Sie dasselbe in der IF-Anweisung sieben Zeilen darunter. Ist es aus irgendeinem Grund für den Rechner nicht möglich, das Fenster (Window) zu öffnen, dann wird das Programm beendet und eine Fehlermeldung ausgegeben.

Dank Etikett können Sie nun Ihre Disketten schnell und einfach beschriften und haben so einen besseren Überblick über den Inhalt.

(Holger Stöter/Rudolf Seitz/rb)

```
1k0 /**************************************
kR1 **    Diskettenetiketten drucken     **
G8  ** für Epson- und NEC P6  Drucker    **
QZ  **         von  Holger Stöter        **
K3  **************************************/
4r0 #include <libraries/dos.h>
GR  #include <stdio.h>
B5  #include <exec/types.h>
Xm  #include <intuition/intuition.h>
RF  #include <intuition/intuitionbase.h>
xK  struct FileInfoBlock fib1;
f8  struct IntuitionBase *IntuitionBase;
G8  struct Window *fenster;
6N  struct IntuiText BText =
Be  {
QL1  0,1,
kC   JAM2,
hr   10,10,
u9   NULL,
vA   NULL,
mn   NULL
wd0 },
Wi  PText =
Kn  {
ZU1  0,1,
tL   JAM2,
x0   5,4,
3I   NULL,
4J   NULL,
vw   NULL
5m0 },
bl  NText =
Tw  {
id1  0,1,
2U   JAM2,
```

Listing. Mit »Etikett« können Sie Ihre Disketten beschriften. Bitte mit dem Checksummer eingeben.

```
69  5,4,
CR  NULL,
DS  NULL,
45  NULL
xt0 };
6w  void exitprg();
1u  request();
HT  char zeile[50], name[40][30], diskname[31], help[30];
Bu  FILE *printer;
dt  char escE[] = {15,27,'S',0,27,'A',5,27,'U',1,10};
qK  char escN[] = {27,'S',0,27,'A',5,27,'g',10};
55  main ()
iB  {
OV1  short test;
E4   long zeiger,type[40],thelp;
Or   int handle,window,i,j=0,t,k,x,y,lines,ptyp;
BX   char *mouse = (char *)0xbfe001, spaces[8];
Tz0 /* Intuition-Library öffnen */
9t1  if (!(IntuitionBase = (struct IntuitionBase *)
Fd7        OpenLibrary("intuition.library",0L)))
xt8        exit (FALSE);
6C0 /* Window öffnen und Meldung ausgeben */
7p1  window =
X1   open("RAW:0/0/640/256/Disk-Etiketten V2.3  (c) 1987 Holger
     Stöter   *Ja* *Nein*",2,0);
b8   if (window == -1)
v0   {
rF2   printf("Ich kann das Window nicht öffnen !!!!\n\n");
Ir    CloseLibrary(IntuitionBase);
Tr    exit(0);
3Y1  }
eS   sprintf(zeile,"    **************************************\n
     ");
Rq   write(window,zeile,strlen(zeile));
id   sprintf(zeile,"    ***  Disketten-Etiketten drucken !  ***\n
     ");
Ts   write(window,zeile,strlen(zeile));
Ae   sprintf(zeile,"    ***              V2.3              ***\n
     ");
Vu   write(window,zeile,strlen(zeile));
Q7   sprintf(zeile,"    ***     (c) 1987  Holger Stöter     ***\n
     ");
Xw   write(window,zeile,strlen(zeile));
AW   sprintf(zeile,"    **************************************\n
     \n");
Zy   write(window,zeile,strlen(zeile));
BH0 /* Zeiger auf aktive Window-Struktur holen */
yz1  fenster = (struct Window *) IntuitionBase->ActiveWindow;
Yq   WindowLimits (fenster,640L,256L,640L,256L);
Sv0 /* Abfrage, ob Epson- oder NEC P6 Drucker */
g51  if(request("Bitte Druckertyp anwählen !"," EPSON "," NEC P6
     "))
Fi   {
552   lines = 29;
t0    strcpy(&spaces,"       ");  /* 7 Spaces */
p1    ptyp=0;
hD    t=0;
Ot1  }
E1   else
Mp   {
Wy2   lines = 22;
hJ    strcpy(&spaces,"   ");       /* 3 Spaces */
yB    ptyp=1;
4e    t=4;
V01  }
Cd0 /* Escape-Sequenz übergeben */
Q51  if(!(request("Bitte Drucker einschalten !"," OK "," EXIT ")
     ))
Ux   {
Rk2   close(window);
rQ    CloseLibrary(IntuitionBase);
2Q    exit(0);
c71  }
pM   handle = open("par:",1,0);
a4   if (handle == -1)
b4   {
f32   printf("Ich kann den Drucker nicht ansprechen !!!!\n\n");
Zs    close(window);
zY    CloseLibrary(IntuitionBase);
AY    exit(0);
kF1  }
v6   if(ptyp)
iB   {
7n2   write(handle,escN,sizeof(escN));
oJ1  }
eR   else
mF   {
1P2   write(handle,escE,sizeof(escE));
sN1  }
Y9   close(handle);
G30 /* Drucker zur Ausgabe öffnen !!!! */
Mv1  if (!(printer = fopen("par:","a")))
sL   {
wK2   printf("Ich kann den Drucker nicht ansprechen !!!!\n\n");
q9    close(window);
WK    exitprg;
OV1  }
A00 /* Directory öffnen */
w41  for(;;)      /* Endlosschleife */
zS   {
2r2   i=t;j=0;
UM    if(!(request("Bitte Diskette einlegen !"," OK "," EXIT "))
      )
2V    {
zI3    close(window);
mg     exitprg();
9e2   }
KN    zeiger = Lock ("df0:",-2);        /* Laufwerk 0 ansprechen
      */
cX    test = Examine (zeiger,&fib1);   /* Diskettennamen holen *
      /
4z    strcpy(&diskname,&fib1.fib_FileName);
Or    sprintf(zeile,"\n\n\n\n\nFolgende Namen enthält die Disket
      te :\n\n");
az    write(window,zeile,strlen(zeile));
yx    while (test)     /* "test" wird 0 bei Directoryende */
Cf    {
Gc3    if (!j) goto label1;
q5     sprintf(zeile,"%-17.17s",fib1.fib_FileName);
f4     write(window,zeile,strlen(zeile));
ft     if (fib1.fib_DirEntryType > 0)
Hk     {
Rm4     sprintf(zeile,"   (dir) ? (klick j/n) ");
j8      write(window,zeile,strlen(zeile));
Ot3    }
E1     else
Mp     {
OL4     sprintf(zeile,"         ? (klick j/n) ");
oD      write(window,zeile,strlen(zeile));
Ty3    }
ay     k=1;
GG     while (k)
Sv     {
zd4     while((*mouse & 0x40)==0x40);   /* Auf linken Mausknopfk
        lick warten */
10      x = fenster->MouseX;             /* Aktuelle x und y Pos
        ition des */
Cc      y = fenster->MouseY;             /* Mauszeigers holen */
Co      if (x>=383 && x<=413 && y>=0 && y<=10)  /* JA angekl
        ickt */
X0      {
zA5      type[i] = fib1.fib_DirEntryType;
Eu       strcpy(&name[i],&fib1.fib_FileName);
8Q       i++;k=0;
YT       sprintf(zeile,"Ja      %d\n",40-i);
gB4     }
P1      if (x>=431 && x<=477 && y>=0 && y<=10)  /* NEIN ange
        klickt */
e7      {
185      k=0;
xR       sprintf(zeile,"Nein    %d\n",40-i);
1G4     }
mH3    }
9Y     write(window,zeile,strlen(zeile));
7j     if (i==40) break;
6R     label1: j++;
Dn     test = ExNext (zeiger,&fib1);   /* Nächsten Directoryname
       n holen */
rM2   }
su    i--;
NT    for (k=0;k<=i;k++)              /* Ausgewählte Namen so s
      ortieren, */
pH    {                               /* daß Unterdirectories ob
      en auf   */
163    for (j=t;j<i;j++)              /* dem Etikett erscheinen
```

Listing. Mit »Etikett« können Sie Ihre Disketten beschriften (Fortsetzung)

```
           .         */
182 sL      {
183 2b4      if (type[j]<type[j+1])
184 uN       {
185 la5       thelp = type[j];
186 8R        type[j] = type[j+1];
187 Jo        type[j+1] = thelp;
188 1G        strcpy(&help, &name[j]);
189 ER        strcpy(&name[j], &name[j+1]);
190 ku        strcpy(&name[j+1], &help);
191 5a4      }
192 6b3     }
193 7c2    }
194 yl     for (k=t;k<=i;k++)             /* Unterdirectories mit "*" k
           ennzeichnen */
195 5Y     {
196 4u3     if (type[k]>0)
197 7a      {
198 yz4      strcpy(&help,"*");
199 TW       strcat(&help,&name[k]);
200 Dz       strcpy(&name[k],&help);
201 Fk3     }
202 G12    }
203 6m     if(!(request("Bitte Etikett einrichten !"," OK "," EXIT ")
           ))
204 Eh     {
205 BU3     close(window);
206 ys      exitprg();
207 Lq2    }
208 rR0 /* Etikett ausdrucken */
209 Pm2    k=0;
210 cg     fprintf(printer,"\n\n\n\n\n\n\n\n");
211 08     fprintf(printer,"%s%-32.32s\n",spaces,diskname);
212 WI     fprintf(printer,"%s-------------------------------------\n"
           ,spaces);
213 8r     for (j=t;j<=i;j+=2)
214 Or     {
215 JJ3     k++;
216 Qv      fprintf(printer,"%s%-18.17s%-17.17s\n",spaces,name[j],nam
            e[j+1]);
217 V02    }
218 Gr     fprintf(printer,"%s-----------------------------------\n\
           n\n",spaces);
219 gp     for (j=0;j<=(lines-k);j++)     /* Bis zum nächsten Etikett
            vorschieben */
220 tu     fprintf(printer,"\n");
221 Z41   }
222 VJ    close(window);     /* PRG-Schluß (nur formal) */
223 F9    exitprg();
224 c70 }
225 Kz  /* Funktionen */
226 7j  /* Funktion um PRG ordnungsgemäß zu beenden */
227 U4  void exitprg()
228 c51  {
229 H42   fclose (printer);
230 zY    CloseLibrary(IntuitionBase);
231 AY    exit(0);
232 kF1  }
233 LS0 /* Funktion um Requester auszugeben */
234 Bj  request (BodyText,PositivText,NegativText)
235 fb1  char *BodyText,*PositivText,*NegativText;
236 kD   {
237 br2   BText.IText = (UBYTE *) BodyText;
238 Hf    PText.IText = (UBYTE *) PositivText;
239 cc    NText.IText = (UBYTE *) NegativText;
240 Q8    if (AutoRequest(NULL,&BText,&PText,&NText,NULL,NULL,370L,6
          0L))
241 pI    {
242 RN3    return(1);
243 vQ2   }
244 1Y    else
245 tM    {
246 RM3    return(0);
247 zU2   }
248 OV1  }
(C) 1988 M&T
```

Listing. Mit »Etikett« können Sie Ihre Disketten beschriften (Schluß). Bitte mit dem Checksummer eingeben.

Pseudo-Guru

Und es geht weiter mit den kleinen, aber teuflischen Fehlern. Aber Sie bekommen ja immer von uns die Lösung geliefert.

Wie Sie jetzt sicherlich schon wissen, verstecken wir in kurzen Listings kleine Fehler. Die Auflösung folgt immer in der nächsten Ausgabe. Die letzte Aufgabe war nicht weiter schwer. Der Aufruf von »scanf« war diesmal richtig. Das liegt daran, daß in C Zeichenketten (Strings) wie Zeiger gehandhabt werden. Es ist also in diesem Fall nicht nötig, den Adreßoperator zu verwenden.

Der Fehler war diesmal ein ganz kleiner, trotzdem wird er oft gemacht. Wie Sie wissen, wird in C normalerweise nach jeder Anweisung ein Strichpunkt gesetzt.

Hüten Sie sich jedoch davor, wie in unserem Beispiel, nach einer for-Anweisung dies zu tun. Der C-Compiler interpretiert das dann als leere Anweisung. Diese wird dann folgerichtig so oft durchgeführt, wie in der Anweisung angegeben. Der eigentliche Schleifenkörper hingegen wird nur einmal durchlaufen. Daher resultiert auch die Ausgabe des ersten Zeichens.

Passen Sie auf solche »Kleinigkeiten« besonders auf, da die Konsequenzen sehr groß sein können. Durch solche Fehler entstehen nämlich auch oft Endlosschleifen. Außerdem sind solche Bugs sehr schwer zu finden.
So nun wieder eine neue Aufgabe, diesmal wieder in Basic.

Wir wollen Speicherbereiche des Amiga ansehen. Die Ausgabe soll wahlweise in Byte, Worten oder Langworten geschehen. Zuerst geben wir aber die untere und obere Grenze des Bereichs an. Hier das entsprechende Listing:

```
Anfang:
PRINT
INPUT "Untere Grenze ";ub&
INPUT "Obere Grenze  ";ob&
INPUT "(B)yte (W)ort (L)angwort ";a$
IF a$="x" THEN END
IF a$<>"b" AND a$><"w" AND a$<>"l" THEN a$="
w"
CALL ZeigeBereich(ub&,ob&,a$)
GOTO Anfang
END

SUB ZeigeBereich(unt&,ob&,mode$) STATIC
z=0
WHILE unt&<ob&
  IF mode$="b" THEN
    PRINT USING "####";PEEK(unt&);
  ELSEIF mode$="w" THEN
    PRINT USING "######";PEEKW(unt&);
  ELSEIF mode$="l" THEN
    PRINT USING "########";PEEKL(unt&);
  END IF
  z=z+1:IF z>5 THEN z=0:PRINT
  unt&=unt&+1
WEND
END SUB
```

Testen Sie das Programm mit verschiedensten Werten. Daß die Ausgabe manchmal funktioniert und manchmal nicht, muß wohl an den Gesetzen des McMurphy liegen. Oder haben Sie eine andere Lösung? (rb)

Bilder-

Jahrelang sammelt man tolle Grafiken für den C 64 und dann steigt man auf den Amiga um. Aber die Bilder sind nicht verloren! »PicTransfer« überträgt die verschiedenen Grafikformate schnell und sicher über den Parallelport.

Viele Aufsteiger vom C 64 haben noch sehr viele Bilder, die Sie gerne auf dem Amiga einsetzen würden. Das Problem ist nur die Übertragung der Grafikdaten auf den Amiga. Die serielle Übertragung ist zwar möglich, aber der C 64 besitzt ja von Haus aus keine RS232-Schnittstelle. Deswegen verwendet PicTransfer den parallelen Weg. Ein einfaches 10adriges Kabel verbindet den User-Port des C 64 mit dem Druckerausgang des Amiga, der ja glücklicherweise auch als Eingang verwendet werden kann.

2 Meter 10adriges Flachbandkabel
24poliger Userportstecker
25poliger Sub-D-Stecker für Amiga 500 und Amiga 2000
25polige Sub-D-Buchse Amiga 1000

Tabelle 1. Die benötigten Bauteile für das Verbindungskabel

Für diese Verbindung benötigen Sie die Bauteile (siehe Tabelle 1) und etwas Bastelgeschick. Die Verdrahtung muß erfolgen wie im Bild zu sehen ist. Achten Sie dabei bitte genau auf die Nummern der Anschlußpins. Bei falschem Anschluß können Schäden an beiden Computern die Folge sein!

Nach dem Zusammenbau des Kabels können Sie dies bei ausgeschalteten Computern einstecken. Dann schalten Sie beide ein und laden das jeweilige Programm (siehe Listing 1 und 2). Zuerst zur Software des C 64 (»Alter vor Schönheit!«). Das Programm »PicTransferC64« besitzt sechs Menüpunkte.

F1: Grafikformat wählen

Es kann zwischen vier verschiedenen Grafikformaten umgeschaltet werden.

- Hi-Eddi (Mono) bezieht sich auf zweifarbige Grafiken mit einer Auflösung von 320 x 200 Punkten. Bilder von Programmen wie Giga-Cad, Printfox etc. lassen sich hiermit übertragen.
- Hi-Eddi (Color) gilt für Bilder mit 320 x 200 Punkten, aber zusätzlicher Farbinformation. Sie werden zum Beispiel von Hi-Eddi im Farbmodus generiert und haben auf der Diskette meist 37 Blocks.
- Koalapainter: Für Multicolor-Bilder (160 x 200 Punkte, 16 Farben). Diese Dateien belegen 40 Blöcke auf der Diskette.
- Blazing Paddles: Dies sind zwar auch Farbbilder, sie belegen aber 41 Blöcke und haben ein anderes Format.

Für alle anderen Formate gibt es Programme, die Sie in eins der oberen umwandeln.

F2 : Directory anzeigen

Hiermit lassen Sie sich das Inhaltsverzeichnis der Diskette anzeigen. Wenn der Bildschirm voll ist, kann durch Tastendruck jeweils eine Zeile nach oben gescrollt werden.

F3 : Bild laden

Bevor Sie ein Bild laden, müssen Sie das richtige Format eingestellt haben, da sonst die Daten falsch interpretiert werden. Die Folge ist meist ein wirres Durcheinander. Bei Bildern von Koalapainter ersetzen Sie beim Laden das erste Zeichen durch ein Fragezeichen.

F5 : Bild zeigen

Das geladene Bild wird angezeigt. Mit einem Tastendruck gelangen Sie wieder ins Hauptmenü. Ein Bild kann nur in dem Format gezeigt werden, in dem Sie es geladen haben. Sollte sich also nach der Betätigung von <F5> nichts tun, überprüfen Sie bitte das Format (Taste F1).

F7 : Übertragung starten

Zunächst erfolgt die Aufforderung, den Amiga empfangsbereit zu machen.

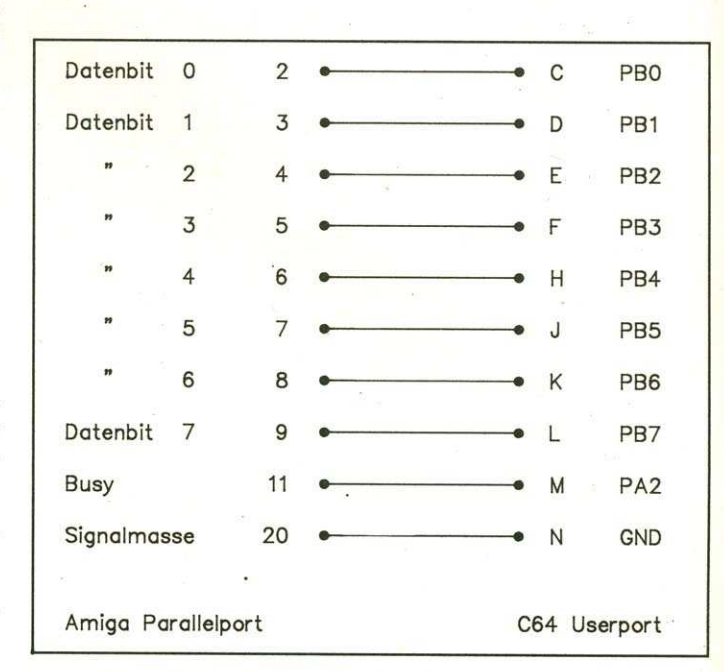
Achtung! Dies darf wirklich erst jetzt geschehen! Nach Drücken von <RETURN> beginnt der eigentliche Transfer.

F8 : Programm beenden

Nach einer Sicherheitsabfrage können Sie Transfer.64 verlassen und normal weiterarbeiten.

Soweit zur Bedienung der Software auf dem C 64. Das Programm »PicTransfer.Amiga« dient zum Empfangen der gesendeten Daten. Der Start erfolgt durch den Aufruf

```
Transfer.Amiga
```



Die notwendigen Verbindungskabel

Es erscheint ein Fenster, in dem die Befehle stehen.

B - Bild empfangen

Dieses Kommando dürfen Sie erst geben, wenn der C 64 Sie auffordert, den Amiga bereitzumachen. Beachten Sie dies nicht, können später Übertragungsfehler auftreten. Nach dem Tastendruck läßt sich der Mauszeiger nicht mehr bewegen. Sobald aber alle Daten empfangen wurden, verhält sich der Zeiger wieder normal wie vor dem Start.

1. Bei ausgeschalteten Computern das Verbindungskabel anschließen.
2. Beide Computer einschalten und die entsprechenden Programme starten.
3. Auf dem C 64 mit <F1> das Grafikformat wählen.
4. Mit <F3> das gewünschte Bild laden.
5. Die Datenübertragung mit <F7> vorbereiten.
6. Beim Amiga <B> eingeben.
7. Beim C 64 <RETURN> drücken, was die Übertragung startet.
8. Nach erfolgtem Transfer auf dem Amiga mit <S> das Bild speichern.

Tabelle 2. Die genaue Vorgehensweise beim Bilderübertragen

Recycling

D - Directory
Hier können Sie sich das Directory anzeigen lassen. Es ist möglich, komplette Pfade einzugeben, zum Beispiel:

```
DF1:Bilder/Schwarzweiß
```

Dieser Befehl ist wichtig, wenn Sie die Namen der schon gespeicherten Bilder vergessen haben und diese nicht aus Versehen überschreiben wollen.

S - Bild speichern
Unter dem nun einzugebenden Dateinamen wird das Bild im IFF-Format gespeichert. Auch hier können wieder komplette Pfadnamen verwendet werden.

Amiga links - M
Durch Drücken der linken Amiga-Taste und <m> schaltet der Amiga auf den Grafikbildschirm um, wo die Zeichnung zu sehen ist.

Amiga links - N
Durch diese Tastenkombination gelangen Sie wieder auf den Workbenchscreen. Dort können Sie dann weiter das Programm benutzen.

E : Programmende
Nach einer Sicherheitsabfrage kehren Sie zurück in das CLI-Fenster, von dem TransferPic.Amiga aufgerufen wurde.

Nun können die Bilder mit Zeichenprogrammen, die das IFF-Format lesen können, weiterverarbeitet werden. Sicher lassen sich viele Bilder noch verschönern, der Amiga bietet ja in dieser Beziehung fantastische Fähigkeiten an. Außerdem können Sie viel Arbeit sparen, wenn Sie zum Beispiel die Construction-Kits von Hi-Eddi übernehmen. Sie brauchen nicht alles noch einmal selbst zu zeichnen, sondern können sofort mit der Arbeit beginnen. (Martin Jahner/rb)

Programmname:	TransferPic.Amiga
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Assembler
Assembler:	Seka

```
Programm : TransferPic.Amiga
----------------------------

  1 KuO ; Grafik - Übertragung vom C 64   2
        1.9.1987 by Martin Jahner
  2 6W  execbase = 4
  3 YH  Disable      = -120
  4 eQ  Enable       = -126
  5 54  SuperState   = -150
  6 NE  UserState    = -156
  7 VG  AllocMem     = -198
  8 fm  FreeMem      = -210
  9 Oz  OpenLibrary  = -552
 10 wK  CloseLibrary = -414
 11 1g  Open    = -30
 12 Zr  Close   = -36
 13 j5  Read    = -42
 14 P5  Write   = -48
 15 4j  Lock    = -84
 16 rB  UnLock  = -90
 17 7a  Examine = -102
 18 3C  ExNext  = -108
 19 yi  IoErr      = -132
 20 G2  DeleteFile = -72
 21 Xx  OpenScreen  = -198
 22 tS  OpenWindow  = -204
 23 Uv  CloseScreen = -66
 24 ge  CloseWindow = -72
 25 tw  ViewPortAddress = -300
 26 Ya  WBenchToFront   = -342
 27 gw  LoadRGB4 = -192
 28 9h  ddra = $bfd200
 29 P1  ddrb = $bfe301
 30 Qq  dra  = $bfd000
 31 f9  drb  = $bfe101
 32 hJ  transfer:
 33 iM3    move.l  execbase,a6
 34 Fz     lea  dosname,a1
 35 ro     move.l  #0,d0
 36 4n     jsr  OpenLibrary(a6)
 37 UT     move.l  d0,dosbase
 38 Z0     lea  intname,a1
 39 vs     move.l  #0,d0
 40 8r     jsr  OpenLibrary(a6)
 41 ts     move.l  d0,intbase
 42 L4     lea  gfxname,a1
 43 zw     move.l  #0,d0
 44 Cv     jsr  OpenLibrary(a6)
 45 Z8     move.l  d0,gfxbase
 46 k9     move.l  #$2804,d0
 47 86     move.l  #0,d1
 48 GC     jsr  AllocMem(a6)
 49 Td     move.l  d0,adresse
 50 cl     bne  memok
 51 yQ     move.l  #$50b,d0
 52 uv     rts
 53 7Z0 memok:
 54 313    move.l  intbase,a6
 55 Zx     lea  screen_defs,a0
 56 J0     jsr  OpenScreen(a6)
 57 VI     move.l  d0,screenhd
 58 4e     bne  screenok
 59 6Y     move.l  #$50b,d0
 60 23     rts
 61 iH0 screenok:
 62 bS3    lea  window_defs,a0
 63 1Q     jsr  OpenWindow(a6)
 64 0V     move.l  d0,windowhd
 65 69     bne  windowok
 66 Df     move.l  #$50b,d0
 67 9A     rts
 68 AJ0 windowok:
 69 I03    move.l  intbase,a6
 70 Ge     move.l  windowhd,a0
 71 3c     jsr  ViewPortAddress(a6)
 72 h5     move.l  d0,viewport
 73 6s     move.l  gfxbase,a6
 74 Qz     move.l  viewport,a0
 75 dJ     lea  colors,a1
 76 22     move.l  #32,d0
 77 oe     jsr  LoadRGB4(a6)
 78 R9     move.l  intbase,a6
 79 mI     jsr  WBenchToFront(a6)
 80 y0     move.l  dosbase,a6
 81 9W     move.l  #conname,d1
 82 70     move.l  #1005,d2
 83 8n     jsr  Open(a6)
 84 76     move.l  d0,conhd
 85 U3     bne  inloop
 86 Xz     move.l  #$50b,d0
 87 TU     rts
 88 Jr0 inloop:
 89 Zc3    move.l  screenhd,a0
 90 X6     move.l  $c0(a0);bitmap0
 91 pT     move.l  $c4(a0),bitmap1
 92 7q     move.l  $c8(a0),bitmap2
 93 Ij     move.l  $cc(a0),bitmap3
 94 xb     move.l  $d0(a0),bitmap4
 95 k2     lea  menutext,a0
 96 I8     bsr  print
 97 FH     move.l  dosbase,a6
 98 vK     move.l  conhd,d1
 99 9J     move.l  #inbuffer,d2
100 DF     move.l  #2,d3
101 Zg     jsr  Read(a6)
102 Jk     tst.l  d0
103 7j     beq  inloop
104 NR     andi.b  #$df,inbuffer
105 ZK     cmpi.b  #'B',inbuffer
106 0S     beq  empfang
107 QS     cmpi.b  #'S',inbuffer
108 Vf     beq  speichern
109 jW     cmpi.b  #'D',inbuffer
110 JE     beq  directory
111 oc     cmpi.b  #'E',inbuffer
112 vU     bne  inloop
113 fe     lea  sicher,a0
114 aQ     bsr  print
115 XZ     move.l  dosbase,a6
116 Dc     move.l  conhd,d1
117 Rb     move.l  #inbuffer,d2
118 VX     move.l  #2,d3
119 ry     jsr  Read(a6)
120 dh     andi.b  #$df,inbuffer
121 D6     cmpi.b  #'J',inbuffer
122 Ch     beq  ende
123 2b     bra  inloop
124 TN0 ende:
125 hj3    move.l  dosbase,a6
126 Nm     move.l  conhd,d1
127 Cp     jsr  Close(a6)
128 Fx     move.l  intbase,a6
129 Db     move.l  windowhd,a0
130 C0     jsr  CloseWindow(a6)
131 FI     move.l  screenhd,a0
132 J2     jsr  CloseScreen(a6)
133 Ky     move.l  execbase,a6
134 0g     move.l  adresse,a1
135 Ba     move.l  #$2804,d0
136 64     jsr  FreeMem(a6)
137 UR     move.l  dosbase,a1
138 8i     jsr  CloseLibrary(a6)
139 1e     move.l  intbase,a1
140 Ak     jsr  CloseLibrary(a6)
```

Listing 1. »TransferPic.Amiga« dient zum Empfangen der Bilder von der parallelen Schnittstelle. Bitte mit dem Checksummer eingeben.

```
nU          move.l  gfxbase,a1
Cm          jsr     CloseLibrary(a6)
bY          move.l  #0,d0
OP          rts
v10 directory:
Zs3         move    #-1,dirspecial
mU          lea     dirtext,a0
8y          bsr     print
57          move.l  dosbase,a6
1A          move.l  conhd,d1
Nf          move.l  #dirname,d2
cT          move.l  #80,d3
PW          jsr     Read(a6)
gI          lea     dirname,a0
kr          move.b  #0,-1(a0,d0)
Mi          move.l  #dirname,d1
L7          move.l  #-2,d2
gR          jsr     Lock(a6)
Ef          tst.l   d0
XJ          beq     direrror1
Vc          move.l  d0,locksave
pf          move.l  locksave,d1
bR          move.l  #fileinfo,d2
A6          jsr     Examine(a6)
Kl          tst.l   d0
zr          beq     direrror
SQ          lea     dirtitel,a0
SI          bsr     print
cK0 dirloop:
gW3         lea     fileinfo,a0
YD          cmp     #-1,dirspecial
e3          bne     dirl1
qJ          addq.l  #2,a0
Ra0 dirl1:
0X3         addq.l  #6,a0
aQ          bsr     print
Fj          lea     return,a0
cS          bsr     print
6w          move.l  locksave,d1
si          move.l  #fileinfo,d2
Y0          jsr     ExNext(a6)
b2          tst.l   d0
om          bne     dirloop
L60 direrror:
C23         move.l  locksave,d1
SE          jsr     UnLock(a6)
B70 direrror1:
by3         jsr     IoErr(a6)
Pt          cmp.l   #$e8,d0
rq          beq     dirend
jf0 diskerror:
RF3         lea     diskerrtext,a0
rh          bsr     print
iu0 dirend:
823         lea     rettext,a0
uk          bsr     print
Wv          move.l  conhd,d1
ku          move.l  #inbuffer,d2
oq          move.l  #2,d3
AH          jsr     Read(a6)
Ir          bra     inloop
wL0 empfang:
A13         lea     empfangtext,a0
2s          bsr     print
Xw          move.b  #0,ddrb
7F          andi.b  #$fe,ddra
WA          move.l  execbase,a6
UZ          jsr     SuperState(a6)
xZ          move.l  d0,usersp
rS          jsr     Disable(a6)
Yp          move.l  adresse,a0
ew          move.l  adresse,a1
nx          add.l   #$2804,a1
Nn0 eloop1:
nu3         btst    #0,dra
qP          beq     eloop1
pw          btst    #0,dra
sR          beq     eloop1
E7          move.b  drb,(a0)+
Vw0 eloop2:
t03         btst    #0,dra
eB          bne     eloop2
v2          btst    #0,dra
gD          bne     eloop2
c0          cmp.l   a1,a0
XG          blt     eloop1
mW          jsr     Enable(a6)
aw          move.l  usersp,d0
i0          jsr     UserState(a6)
Uy          bsr     pruefsumme
s9          move.l  adresse,a0
8t          move.b  $2800(a0),formcode
HT          move.b  $2801(a0),c64summe+3
IU          move.b  $2802(a0),c64summe+2
JV          move.b  $2803(a0),c64summe+1
UX          move.l  c64summe,d0
0S          cmp.l   summe,d0
ae          beq     summeok
JH          lea     fehlertext,a0
cS          bsr     print
Zb          move.l  dosbase,a6
Fe          move.l  conhd,d1
Td          move.l  #inbuffer,d2
XZ          move.l  #2,d3
t0          jsr     Read(a6)
1a          bra     inloop
U80 summeok:
iF3         move.b  #0,hinter
AR          move.l  adresse,a0
aX          move.b  formcode,d0
7G          cmp.b   #0,d0
pK          beq     form0
FK          cmp.b   #1,d0
tP          beq     form1
N0          cmp.b   #2,d0
xU          beq     form2
VS          cmp.b   #3,d0
1Z          beq     form3
En          bra     inloop
Ma0 form2:
n03         move.b  $2710(a0),hinter
e6          or.b    #$10,hinter
T1          move.l  adresse,a1
Zs          move.l  adresse,a2
nx          add.l   #$2328,a0
8A          add.l   #$2400,a1
0V          add.l   #$2800,a2
An          bsr     copy
U1          move.l  adresse,a0
as          move.l  adresse,a1
gz          move.l  adresse,a2
ph          add.l   #$1f40,a0
31          add.l   #$2000,a1
JM          add.l   #$2400,a2
Hu          bsr     copy
is0 form3:
Uc3         bsr     multicolor
X6          bra     inloop
Tp0 form0:
as  form1:
DW3         bsr     singlecolor
bA          bra     inloop
8U0 singlecolor:
J03         move.l  adresse,a0
9V          move.l  adresse,a5
IF          add.l   #$1f40,a5
hI          lea     $2000(a0),a1
7L0 si2:
uo3         move.l  bitmap0,vergleich
1D0 si0:
Yc3         move.l  #0,d7
7K0 si1:
sk3         move.b  (a0)+,d0
0Q          bsr     getsin
pD          bsr     colsin
xX          bsr     bittobyte
rV          bsr     tobitmap
ap          add.l   #40,d7
H0          cmp.l   #320,d7
Ab          blt     si1
xQ          addq.l  #1,a1
1Z          addq.l  #1,bitmap0
rg          addq.l  #1,bitmap1
xn          addq.l  #1,bitmap2
3u          addq.l  #1,bitmap3
91          addq.l  #1,bitmap4
1J          move.l  bitmap0,a3
Ec          sub.l   vergleich,a3
W1          cmp.l   #40,a3
Ff          blt     si0
b9          add.l   #280,bitmap0
iC          add.l   #280,bitmap1
pF          add.l   #280,bitmap2
wI          add.l   #280,bitmap3
3L          add.l   #280,bitmap4
Tb          cmp.l   a5,a0
Wy          blt     si2
CD          rts
Wz0 multicolor:
Ja3         move.l  adresse,a0
j5          move.l  adresse,a5
sp          add.l   #$1f40,a5
Hs          lea     $2000(a0),a1
gM          lea     $2400(a0),a2
6Q0 mu2:
VP3         move.l  bitmap0,vergleich
0I0 mu0:
9D3         move.l  #0,d7
6P0 mu1:
TL3         move.b  (a0)+,d0
JH          bsr     getmul
84          bsr     colmul
Y8          bsr     bittobyte
S6          bsr     tobitmap
BQ          add.l   #40,d7
sz          cmp.l   #320,d7
Fm          blt     mu1
Y1          addq.l  #1,a1
e8          addq.l  #1,a2
NB          addq.l  #1,bitmap0
TI          addq.l  #1,bitmap1
ZP          addq.l  #1,bitmap2
fW          addq.l  #1,bitmap3
1d          addq.l  #1,bitmap4
Nv          move.l  bitmap0,a3
qE          sub.l   vergleich,a3
8d          cmp.l   #40,a3
Lr          blt     mu0
D1          add.l   #280,bitmap0
Ko          add.l   #280,bitmap1
Rr          add.l   #280,bitmap2
Yu          add.l   #280,bitmap3
fx          add.l   #280,bitmap4
5D          cmp.l   a5,a0
cA          blt     mu2
op          rts
Su0 tobitmap:
a83         move.l  bitmap0,a3
H0          move.b  byte0,(a3,d7)
eD          move.l  bitmap1,a3
0W          move.b  byte1,(a3,d7)
iI          move.l  bitmap2,a3
Ve          move.b  byte2,(a3,d7)
mN          move.l  bitmap3,a3
cm          move.b  byte3,(a3,d7)
qS          move.l  bitmap4,a3
ju          move.b  byte4,(a3,d7)
01          rts
0X0 bittobyte:
GD3         move.l  #0,d0
MK          move.l  #0,d1
yT          lea     bit0,a3
Iu          lea     byte0,a4
7D0 bibyloop:
MI3         move.b  (a3,d1),d2
PR          move.b  (a4,d0),d3
ak          roxr.b  #1,d2
gr          roxr.b  #1,d3
```

Listing 1. »TransferPic.Amiga« dient zum Empfangen der Bilder von der parallelen Schnittstelle. Bitte mit dem Checksummer eingeben. (Fortsetzung)


Wenn mal was nicht funktioniert...
Computer-Service
commodore
Schneider
ATARI
Installation
Wartung
Reparatur
RAT UND TAT
Service · Beratung
Technischer Kundendienst
Im gesamten Bundesgebiet vertreten
70 Niederlassungen
Hotline Bereich Nord (040) 2201913
Hotline Bereich Mitte (0201) 35923
Hotline Bereich Süd (08165) 74220
Hotline Berlin (030) 6846057–9

```
rn      move.b  d2,(a3,d1)
wo      move.b  d3,(a4,d0)
Rx      addq.l  #1,d1
22      cmp.l   #8,d1
4L      blt   bibyloop
ZX      move.l   #0,d1
Qv      addq.l   #1,d0
kv      cmp.l   #5,d0
8P      blt   bibyloop
KL      rts
ha0 colsin:
G13     lea  bit0,a3
bY      move.l   #0,d0
RK0 sinloop:
TS3     move.b   (a3,d0),d1
ag      cmp.b   #1,d1
dr      beq   sin0
fs      move.b   (a1),d1
qv      andi.b   #$0f,d1
Bu      or.b   #$10,d1
OV      bra   sin1
Vr0 sin0:
kx3     move.b   (a1),d1
it      lsr.b   #4,d1
Gz      or.b   #$10,d1
e10 sin1:
B53     move.b   d1,(a3,d0)
1G      addq.l   #1,d0
NM      cmp.l   #8,d0
i4      blt   sinloop
fg      rts
ed0 colmul:
b63     lea   bit0,a3
wt      move.l   #0,d0
Q70 colloop:
on3     move.b   (a3,d0),d1
pz      cmp.b   #0,d1
QS      beq   col0
x3      cmp.b   #1,d1
YW      beq   col1
57      cmp.b   #2,d1
ga      beq   col2
7L      move.b   (a2),d1
ZI      or.b   #$10,d1
QD      bra   col3
Xh0 col0:
YG3     move.b   hinter,d1
TG      bra   col3
fq0 col1:
B03     move.b   (a1),d1
9K      lsr.b   #4,d1
hQ      or.b   #$10,d1
YL      bra   col3
p10 col2:
GT3     move.b   (a1),d1
RW      andi.b   #$0f,d1
mV      or.b   #$10,d1
yB0 col3:
hb3     move.b   d1,(a3,d0)
Hm      addq.l   #1,d0
ts      cmp.l   #8,d0
gq      blt   colloop
BC      rts
id0 getsin:
7c3     lea   bit0,a3
XV      move.l   #0,d1
cm0 gs0:
4b3     clr.b   d2
Yg      roxr.b   #1,d0
FJ      roxl.b   #1,d2
zv      move.b   d2,(a3,d1)
Y4      addq.l   #1,d1
99      cmp.l   #8,d1
bz      blt   gs0
NO      rts
WX0 getmul:
Jo3     lea   bit0,a3
jh      move.l   #0,d1
Wa0 gm0:
Jr3     roxr.b   #1,d0
u4      roxr.b   #1,d2
1t      roxr.b   #1,d0
w6      roxr.b   #1,d2
w0      lsr.b   #6,d2
D9      move.b   d2,(a3,d1)
zl      move.b   d2,1(a3,d1)
pM      addq.l   #2,d1
00      cmp.l   #8,d1
Sk      blt   gm0
cd      rts
Po0 copy:
3S3     move.b   (a0)+,(a1)+
k5      cmp.l   a2,a1
Fn      blt   copy
hi      rts
VF0 pruefsumme:
1S3     move.l   #0,summe
p6      move.l   adresse,a0
vD      move.l   adresse,a1
pw      add.l   #$2801,a1
iq0 summeloop:
OP3     clr.l   d0
um      move.b   (a0)+,d0
ql      add.l   d0,summe
mA      cmp.l   a1,a0
Jf      blt   summeloop
tu      rts
yk0 speichern:
1y3     bsr   codierung
at      lea   speichtext,a0
dT      bsr   print
ac      move.l   dosbase,a6
Gf      move.l   conhd,d1
fv      move.l   #filename,d2
7y      move.l   #80,d3
ul      jsr   Read(a6)
61      lea   filename,a0
FM      move.b   #0,-1(a0,d0)
L5      lea   colors,a0
qx      lea   cmap,a1
up      move.l   #31,d1
bY0 colorloop:
MD3     move   (a0),d0
Mg      lsr   #4,d0
KJ      andi   #$f0,d0
wa      move.b   d0,(a1)+
QH      move   (a0),d0
NM      andi   #$f0,d0
zd      move.b   d0,(a1)+
YI      move   (a0)+,d0
5J      lsl   #4,d0
RQ      andi   #$f0,d0
3h      move.b   d0,(a1)+
7s      dbra   d1,colorloop
yD      move.l   #filename,d1
A4      move.l   #1006,d2
61      jsr   Open(a6)
zQ      tst.l   d0
mm      beq   diskerror
7I      move.l   d0,filehd
1h      move.l   filehd,d1
z0      move.l   #header,d2
oy      move.l   #152,d3
XQ      jsr   Write(a6)
HF      cmp.l   #-1,d0
8y      beq   writeerr
ic      move.l   bitmap0,vergleich
k0      add.l   #8000,vergleich
kw0 writeloop1:
NR3     move.l   #0,d7
o00 writeloop:
bY3     lea   bitmap0,a5
Dt      move.l   filehd,d1
td      move.l   (a5,d7),d2
dQ      move.l   #40,d3
jc      jsr   Write(a6)
TR      cmp.l   #-1,d0
KA      beq   writeerr
XC      addq.l   #4,d7
XH      cmp.l   #20,d7
YP      blt   writeloop
b2      add.l   #40,bitmap0
h9      add.l   #40,bitmap1
nG      add.l   #40,bitmap2
tN      add.l   #40,bitmap3
zU      add.l   #40,bitmap4
Y6      move.l   bitmap0,d0
1U      cmp.l   vergleich,d0
EM      blt   writeloop1
UA      move.l   filehd,d1
Od      jsr   Close(a6)
vU      bra   inloop
P00 writeerr:
YE3     move.l   filehd,d1
4h      jsr   Close(a6)
ds      move.l   #filename,d1
qa      jsr   DeleteFile(a6)
1a      bra   inloop
VY0 print:
gi3     move.l   dosbase,a6
Ml      move.l   conhd,d1
UT      move.l   a0,d2
450 printloop:
Kr3     tst.b   (a0)+
dn      bne   printloop
ON      subq.l   #1,a0
Ry      sub.l   d2,a0
ff      move.l   a0,d3
G9      jsr   Write(a6)
EF      rts
dB0 dosname:
4U3     dc.b   'dos.library',0
LD      even
uM0 dosbase:
kd3     dc.l   0
tW0 intname:
y63     dc.b   'intuition.library',0
QI      even
Ah0 intbase:
pi3     dc.l   0
mJ0 gfxname:
kN3     dc.b   'graphics.library',0
VN      even
3U0 gfxbase:
un3     dc.l   0
fT0 conhd:
wp3     dc.l   0
8G0 screen_defs:
Nb3     dc.w   0,0
S0      dc.w   320,256
DM      dc.w   5
5n      dc.b   1,3
39      dc.w   2
h7      dc.w   15
4x      dc.l   0
1b      dc.l   screentitle
6z      dc.l   0
70      dc.l   0
zA0 screentitle:
v33     dc.b   'Kein Bild geladen',0
nf      even
hP0 window_defs:
bp3     dc.w   0,0
sF      dc.w   320,200
I0      dc.b   1,3
ML      dc.l   0,$11800
mp      dc.l   0,0
HA      dc.l   0
j40 screenhd:
JC3     dc.l   0
KD      dc.l   0
ON      dc.w   320,200
10      dc.w   320,200
1R      dc.w   15
A50 windowhd:
PI3     dc.l   0
T10 viewport:
RK3     dc.l   0
3V0 conname:
lp3     dc.b   'CON:0/0/640/200/** Graf
        ik-Übertragung vom C 64   '
6n4     dc.b   ' 21.9.1987 by Martin J
        ahner **',0
803     even
z50 adresse:
XQ3     dc.l   0
```

```
623 XwO locksave:
624 ZS3    dc.l  0
625 vnO menutext:
626 Ui3    dc.b   12,10,10
627 OT     dc.b   '          B - Bild emp
           fangen',13,10,10
628 Qr     dc.b   '          D - Director
           y',13,10,10
629 qC     dc.b   '          S - Bild spe
           ichern',13,10,10
630 Lb     dc.b   '     Amiga-M - Bild zei
           gen',13,10,10
631 yt     dc.b   '     Amiga-N - Menü zei
           gen',13,10,10
632 MG     dc.b   '          E - Programm
           ende',13,10,10,10
633 w3     dc.b   '         Bitte die Eing
           abe --> '
634 ve     dc.b   0
635 FaO inbuffer:
636 TM3    dc.b   0,0
637 PH     even
638 QkO colors:
639 kC3    dc.w   $000,$eca,$e00,$a00,$d80,
           $fe0,$8f0,$080
640 Ru     dc.w   $0b6,$0dd,$0af,$07c,$00f,
           $70f,$c0e,$c08
641 1W     dc.w   $000,$fff,$b34,$0da,$a0a,
           $5b5,$44a,$ff5
642 8f     dc.w   $b60,$752,$e78,$554,$887,
           $9f9,$88f,$bba
643 qOO empfangtext:
644 643    dc.b   13,10,10,10
645 sr     dc.b   '         >> Bild wird em
           pfangen << ',0
646 h50 fehlertext:
647 jm3    dc.b   13
648 DI     dc.b   '          !!! Übertragungsf
           ehler !!! ',0
649 WDO dirtext:
650 xC3    dc.b   13,10,10
651 MS     dc.b   '      Directory-Name : '
           ,0
652 RUO sicher:
653 FD3    dc.b   13,10,10,10
654 Bm     dc.b   '         Sicher ??? ',0
655 X50 return:
656 MA3    dc.b   13,10
657 x8     dc.b   '     ',0
658 3wO rettext:
659 6L3    dc.b   13,10,10
660 1D     dc.b   '       < Return > ',0
661 Y80 diskerrtext:
662 SG3    dc.b   13,10
663 eb     dc.b   '      Diskettenfehler !!
           ! ',0
664 zdO speichtext:
665 CR3    dc.b   13,10,10
666 MP     dc.b   '       Filename : ',0
667 wNO filename:
668 QF3    blk.b  80
669 h30 dirtitel:
670 VI3    dc.b   12,10
671 RC     dc.b   '   Directory von ',0
672 wNO dirname:
673 VK3    blk.b  80
674 Os     even
675 Z50 filehd:
676 PI3    dc.l   0
677 zhO dirspecial:
678 Vn3    dc.w   -1
679 NOO fileinfo:
680 Ep3    blk.b   260
681 BoO c64summe:
682 VO3    dc.l   0
683 VfO summe:
684 XQ3    dc.l   0
685 V30 formcode:
686 1U3    dc.b   0
687 KZO hinter:
688 nW3    dc.b   0
689 F7     even
690 kMO bitmap0:
691 eX3    dc.l   0
692 pSO bitmap1:
693 gZ3    dc.l   0
694 uYO bitmap2:
695 ib3    dc.l   0
696 zeO bitmap3:
697 kd3    dc.l   0
698 4kO bitmap4:
699 mf3    dc.l   0
700 BMO bit0:
701 Oj3    dc.b   0
702 IUO bit1:
703 2l3    dc.b   0
704 PcO bit2:
705 4n3    dc.b   0
706 WkO bit3:
707 6p3    dc.b   0
```

Listing 1. »TransferPic.Amiga« dient zum Empfangen der Bilder von der parallelen Schnittstelle. Bitte mit dem Checksummer eingeben. (Fortsetzung)

```
708 ds0 bit4:
709 8r3    dc.b   0
710 k00 bit5:
711 At3    dc.b   0
712 r80 bit6:
713 Cv3    dc.b   0
714 yG0 bit7:
715 Ex3    dc.b   0
716 Mi0 byte0:
717 Gz3    dc.b   0
718 Um0 byte1:
719 I13    dc.b   0
720 cq0 byte2:
721 K33    dc.b   0
722 ku0 byte3:
723 M53    dc.b   0
724 sy0 byte4:
725 073    dc.b   0
726 qi     even
727 cX0 vergleich:
728 F83    dc.l   0
729 Sx0 usersp:
730 HA3    dc.l   0
731 iQ0 header:
732 aY3    dc.l   'FORM'
733 RI0 formlen:
734 9u3    dc.l   40144
735 41     dc.l   'ILBM'
736 N6     dc.l   'BMHD'
737 øz     dc.l   20
738 4R     dc.w   320,200
739 p3     dc.w   0,0
740 xl     dc.b   5
741 vS     dc.b   0,0,0,0,0 ; 0,1,0,0,0
742 YK     dc.b   10,11
743 9W     dc.w   320,200
744 wl     dc.l   'CMAP'
745 sG     dc.l   96
746 540 cmap:
747 Mo3    blk.b   96
748 eW     dc.l   'BODY'
749 yf     dc.l   40000
750 4f0 codierung:
751 BC3    rts
(C) 1988 M&T
```

Listing 1. »TransferPic.Amiga« dient zum Empfangen der Bilder von der parallelen Schnittstelle. Bitte mit dem Checksummer eingeben. (Schluß)

```
Name : transfer.64          0801 0edd
-------------------------------------
0801 : 0c 08 c3 07 9e 20 32 30  f7
0809 : 36 32 00 00 00 a9 07 a2  07
0811 : 0a a0 0b 8d 86 02 8e 20  d3
0819 : d0 8c 21 d0 a9 ff 85 02  46
0821 : a9 41 a0 0c 20 1e ab a9  09
0829 : 0c a0 0d 20 1e ab 20 23  d3
0831 : 0b 20 3f 0b a0 00 a2 14  3a
0839 : 84 fb 86 fc a9 00 91 fb  d5
0841 : c8 d0 f9 e6 fc a5 fc c9  51
0849 : 48 d0 f1 a9 85 d0 03 20  d6
0851 : e4 ff f0 fb c9 85 d0 03  02
0859 : 4c af 08 c9 86 d0 03 4c  4c
0861 : d5 08 c9 87 d0 03 4c 86  01
0869 : 09 c9 88 d0 03 4c 7e 0a  34
0871 : c9 89 d0 03 4c 43 0e c9  3e
0879 : 8c d0 d4 a2 13 20 ff e9  fd
0881 : a2 13 a0 05 18 20 f0 ff  bc
0889 : a9 e7 a0 0d 20 1e ab 20  d1
0891 : 3b 0c c9 4a f0 0c c9 4e  c1
0899 : d0 f5 a2 13 20 ff e9 4c  b1
08a1 : 50 08 4c e2 fc ff 9b ab  fa
08a9 : bb cb 0d 0d 0d 0d ee a6  71
08b1 : 08 ad a6 08 c9 04 d0 05  44
08b9 : a9 00 8d a6 08 18 a2 07  74
08c1 : a0 11 20 f0 ff ae a6 08  30
08c9 : bd a7 08 bc ab 08 20 1e  ab
08d1 : ab 4c 50 08 a2 13 20 ff  fb
08d9 : e9 18 a2 13 a0 05 20 f0  6e
08e1 : ff a9 db a0 0d 20 1e ab  61
08e9 : a2 00 86 fb a9 00 85 d4  07
08f1 : a9 a6 20 d2 ff a9 9d 20  54
08f9 : d2 ff 20 3b 0c c9 0d f0  5f
0901 : 3e c9 14 f0 28 c9 20 90  b9
0909 : f1 c9 60 b0 ed 20 d2 ff  38
0911 : a6 fb 9d 00 60 e8 e0 10  0d
0919 : 90 d0 20 3b 0c c9 0d f0  a6
0921 : 1e c9 14 d0 f5 20 d2 ff  ef
0929 : a2 0f 4c eb 08 20 d2 ff  b0
0931 : a6 fb ca 10 b5 a9 20 20  f3
0939 : d2 ff a2 00 4c eb 08 a9  4b
0941 : 20 20 d2 ff a5 fb 85 fe  74
0949 : a2 00 a0 60 20 bd ff a9  63
0951 : 02 a2 08 a0 00 20 ba ff  a6
0959 : a9 00 a2 00 a0 20 20 d5  e2
0961 : ff 20 e0 0b a5 fe f0 1a  54
0969 : ad a6 08 85 02 d0 13 a2  54
0971 : 00 a9 f0 9d 00 40 9d 00  ae
0979 : 41 9d 00 42 9d 00 43 e8  8a
0981 : d0 ef 4c 50 08 ad a6 08  ff
0989 : c5 02 f0 03 4c 50 08 c9  e7
0991 : 00 f0 0b c9 01 f0 07 c9  4d
0999 : 02 f0 36 4c 2b 0a a2 00  b8
09a1 : bd 00 40 9d 00 04 bd 00  39
09a9 : 41 9d 00 05 bd 00 42 9d  7a
09b1 : 00 06 bd 00 43 9d 00 07  53
09b9 : e8 d0 e5 ad 18 d0 09 08  75
09c1 : 8d 18 d0 ad 11 d0 09 20  40
09c9 : 8d 11 d0 20 3b 0c 4c a8  ae
09d1 : 0b a2 00 bd 40 3f 9d 00  5a
09d9 : 04 bd 40 40 9d 00 05 bd  3d
09e1 : 40 41 9d 00 06 bd 40 42  fd
09e9 : 9d 00 07 bd 28 43 9d 00  13
09f1 : d8 bd 28 44 9d 00 d9 bd  f7
09f9 : 28 45 9d 00 da bd 28 46  f4
0a01 : 9d 00 db e8 d0 cd ad 18  15
0a09 : d0 09 08 8d 18 d0 ad 11  f2
0a11 : d0 09 20 8d 11 d0 ad 16  9a
0a19 : d0 09 10 8d 16 d0 ad 10  e2
0a21 : 47 8d 21 d0 20 3b 0c 4c  36
0a29 : a8 0b a2 00 bd 00 40 9d  18
0a31 : 00 04 bd 00 41 9d 00 05  ae
0a39 : bd 00 42 9d 00 06 bd 00  62
0a41 : 43 9d 00 07 bd 00 44 9d  5c
0a49 : 00 d8 bd 00 45 9d 00 d9  1a
0a51 : bd 00 46 9d 00 da bd 00  21
0a59 : 47 9d 00 db e8 d0 cd ad  92
0a61 : 18 d0 09 08 8d 18 d0 ad  5d
0a69 : 11 d0 09 20 8d 11 d0 ad  29
0a71 : 16 d0 09 10 8d 16 d0 20  41
0a79 : 3b 0c 4c a8 0b a2 13 20  35
0a81 : ff e9 18 a2 13 a0 02 20  4e
0a89 : f0 ff a9 1e a0 0e 20 1e  de
0a91 : ab 20 31 0b 20 3f 0b 20  63
0a99 : 3b 0c c9 0d d0 f9 20 56  f8
0aa1 : 0b a2 13 20 ff e9 18 a2  bb
0aa9 : 13 a0 05 20 f0 ff a9 f2  ed
0ab1 : a0 0d 20 1e ab a9 20 85  37
0ab9 : fc a9 00 85 fb a0 00 20  40
0ac1 : 3f 0b b1 fb 8d 01 dd 20  0a
0ac9 : 48 0b c8 d0 f2 e6 fc a5  89
0ad1 : fc c9 48 d0 ea 20 3f 0b  a1
0ad9 : ad a6 08 8d 01 dd 20 48  9d
0ae1 : 0b 20 3f 0b ad 00 70 8d  e5
0ae9 : 01 dd 20 48 0b 20 3f 0b  af
0af1 : ad 01 70 8d 01 dd 20 48  fd
0af9 : 0b 20 3f 0b ad 02 70 8d  0d
0b01 : 01 dd 20 48 0b 20 3f 0b  c7
0b09 : 20 23 0b a2 13 20 ff e9  d8
0b11 : 18 a2 13 a0 05 20 f0 ff  68
0b19 : a9 09 a0 0e 20 1e ab 4c  6b
0b21 : 50 08 a9 00 8d 03 dd ad  a3
0b29 : 02 dd 29 fb 8d 02 dd 60  05
0b31 : a9 ff 8d 03 dd ad 02 dd  ad
0b39 : 09 04 8d 02 dd 60 ad 00  80
0b41 : dd 29 fb 8d 00 dd 60 ad  2f
0b49 : 00 dd 09 04 8d 00 dd a2  90
0b51 : ff ca d0 fd 60 a9 20 85  88
0b59 : fc a9 00 85 fb a0 00 8c  b8
0b61 : 00 70 8c 01 70 8c 02 70  31
0b69 : 18 b1 fb 6d 00 70 8d 00  c0
0b71 : 70 ad 01 70 69 00 8d 01  d5
0b79 : 70 ad 02 70 69 00 8d 02  1f
0b81 : 70 c8 d0 e4 e6 fc a5 fc  0d
0b89 : c9 48 d0 dc 18 ad a6 08  e0
0b91 : 6d 00 70 8d 00 70 ad 01  08
0b99 : 70 69 00 8d 01 70 ad 02  be
0ba1 : 70 69 00 8d 02 70 60 ad  f8
0ba9 : 11 d0 29 df 8d 11 d0 ad  69
0bb1 : 16 d0 29 ef 8d 16 d0 ad  a0
0bb9 : 18 d0 29 f7 8d 18 d0 a9  b3
0bc1 : 07 a2 0a a0 0b 8d 86 02  eb
0bc9 : 8e 20 d0 8c 21 d0 a9 41  ef
0bd1 : a0 0c 20 1e ab a9 0c a0  bd
0bd9 : 0d 20 1e ab 4c be 08 a9  22
0be1 : 00 20 bd ff a9 0f a2 08  0e
0be9 : a0 0f 20 ba ff 20 c0 ff  74
0bf1 : a2 0f 20 c6 ff a2 00 86  1e
0bf9 : fb 20 cf ff a6 fb 9d 00  b9
0c01 : 60 e8 c9 0d d0 f1 a9 00  2d
0c09 : 9d 00 60 20 cc ff a9 0f  54
0c11 : 20 c3 ff a2 13 20 ff e9  6d
0c19 : 18 a2 13 a0 05 20 f0 ff  70
0c21 : a9 00 a0 60 20 1e ab ad  fb
0c29 : 00 60 c9 30 d0 07 ad 01  d0
0c31 : 60 c9 30 f0 04 a9 00 85  39
0c39 : fe 60 20 e4 ff f0 fb 60  44
0c41 : 93 08 2a 2a 2a 2a 2a 2a  99
0c49 : 2a 2a 2a 2a 2a 2a 2a 2a  49
0c51 : 2a 2a 2a 2a 2a 2a 2a 2a  51
0c59 : 2a 2a 2a 2a 2a 2a 2a 2a  59
0c61 : 2a 2a 2a 2a 2a 2a 2a 2a  61
0c69 : 2a 2a 2a 20 20 20 20 20  fb
0c71 : 20 20 20 20 20 20 20 20  71
0c79 : 20 20 20 20 20 20 20 20  79
0c81 : 20 20 20 20 20 20 20 20  81
0c89 : 20 20 20 20 20 20 20 20  89
0c91 : 20 2a 2a 20 20 20 20 47  67
0c99 : 52 41 46 49 4b 2d 55 45  45
0ca1 : 42 45 52 54 52 41 47 55  9c
0ca9 : 4e 47 20 5a 55 4d 20 41  b1
0cb1 : 4d 49 47 41 20 20 20 20  61
0cb9 : 20 2a 2a 20 20 20 20 20  41
0cc1 : 20 20 20 20 20 20 20 20  c1
0cc9 : 20 20 20 20 20 20 20 20  c9
0cd1 : 20 20 20 20 20 20 20 20  d1
0cd9 : 20 20 20 20 20 20 20 20  d9
0ce1 : 20 2a 2a 2a 2a 2a 2a 2a  d7
0ce9 : 2a 2a 2a 2a 2a 2a 2a 2a  e9
0cf1 : 2a 2a 2a 2a 2a 2a 2a 2a  f1
0cf9 : 2a 2a 2a 2a 2a 2a 2a 2a  f9
0d01 : 2a 2a 2a 2a 2a 2a 2a 2a  01
0d09 : 2a 2a 00 0d 0d 20 20 20  7d
0d11 : 20 20 12 46 31 92 20 46  43
0d19 : 4f 52 4d 41 54 20 3a 0d  56
0d21 : 0d 20 20 20 20 20 12 46  22
0d29 : 32 92 20 44 49 52 45 43  f8
0d31 : 54 4f 52 59 0d 0d 20 20  e7
0d39 : 20 20 20 12 46 33 92 20  3c
0d41 : 42 49 4c 44 20 4c 41 44  b5
0d49 : 45 4e 0d 0d 20 20 20 20  5e
0d51 : 20 12 46 35 92 20 42 49  78
0d59 : 4c 44 20 5a 45 49 47 45  61
0d61 : 4e 0d 0d 20 20 20 20 20  41
0d69 : 12 46 37 92 20 55 45 42  05
0d71 : 45 52 54 52 41 47 55 4e  7f
0d79 : 47 20 53 54 41 52 54 45  b2
0d81 : 4e 0d 0d 20 20 20 20 20  61
0d89 : 12 46 38 92 20 50 52 4f  8b
0d91 : 47 52 41 4d 4d 45 4e 44  bc
0d99 : 45 00 48 49 2d 45 44 44  b0
0da1 : 49 20 28 4d 4f 4e 4f 29  a5
0da9 : 20 00 48 49 2d 45 44 44  9b
0db1 : 49 20 28 43 4f 4c 4f 52  b6
0db9 : 29 00 4b 4f 41 4c 41 50  bb
0dc1 : 41 49 4e 54 45 52 20 20  6d
0dc9 : 20 00 42 4c 41 5a 49 4e  ac
0dd1 : 47 20 50 41 44 44 4c 45  87
0dd9 : 53 00 46 49 4c 45 4e 41  92
0de1 : 4d 45 20 3a 20 00 53 49  02
0de9 : 43 48 45 52 20 3f 3f 3f  63
0df1 : 00 55 45 42 45 52 54 52  12
0df9 : 41 47 55 4e 47 20 47 45  1a
0e01 : 53 54 41 52 54 45 54 00  da
0e09 : 55 45 42 45 52 54 52 41  ce
0e11 : 47 55 4e 47 20 42 45 45  33
0e19 : 4e 44 45 54 00 42 49 54  45
0e21 : 54 45 20 41 4d 49 47 41  07
0e29 : 20 42 45 52 45 49 54 4d  91
0e31 : 41 43 48 45 4e 20 2e 2e  ca
0e39 : 2e 28 52 45 54 55 52 4e  8e
0e41 : 29 00 a9 93 20 d2 ff 18  10
0e49 : a2 00 a0 07 20 f0 ff a9  d1
0e51 : 24 85 fe a9 01 a2 fe a0  8f
0e59 : 00 20 bd ff a9 0f a2 08  86
0e61 : a0 60 20 ba ff 20 c0 ff  95
0e69 : b0 54 a2 0f 20 c6 ff 20  46
0e71 : cf ff 20 cf ff 20 cf ff  82
0e79 : 20 cf ff 20 b7 ff d0 3e  c0
0e81 : 20 cf ff 85 fb 20 cf ff  39
0e89 : 85 fc 20 d5 0e a9 20 20  3e
0e91 : d2 ff 20 cf ff 20 d2 ff  b1
0e99 : d0 f8 38 20 f0 ff e0 18  ba
0ea1 : 90 08 a9 00 85 c6 a5 c6  52
0ea9 : f0 fc a9 0d 20 d2 ff a2  01
0eb1 : 00 a9 20 20 d2 ff e8 e0  24
0eb9 : 07 d0 f6 4c 76 0e 20 cc  61
0ec1 : ff a9 0f 20 c3 ff 38 20  ba
0ec9 : f0 ff e0 18 b0 03 20 3b  0e
0ed1 : 0c 4c a8 0b a5 fc a6 fb  64
0ed9 : 20 cd bd 60 01 20 c3 ff  7b
```

Listing 2. Mit »Transfer.64« übertragen Sie Bilder auf den Amiga. Bitte mit dem MSE auf dem C64 eingeben.



C.S.S.

Amiga 500	1098,—	Amiga 2000	2249,—
Monitor 1084	659,—	Philips Stereo	669,—
NEC Multisync, dt.	1359,—	Multi, 15″, 850x650	1759,—

Komplett-Systeme bitte Preise erfragen!

Lw 3½″ intern 269,— extern, durchg. Bus 349,—
Lw 5¼″ extern 80 Track 1.2 MB 349,—

A 500 Speichererw. 512 K, abschaltbar nur 249,—
A 2000 Speicherkarte, 2 MB, autokonfigurierend, nur 898,—
A 2000 PC-Karte m. 5¼″ Lw u. MS-DOS 3.2, GW-Basic 1098,—
A 2000 AT-Karte m. 1.2 MB Lw u. MS-DOS 3.2, GW-Basic 1859,—
A 2000 Turbo Karte, CPU 68020 / 68881, 14 MHz 3299,—
Festplatten von 20 — 120 MB lieferbar; bitte anfragen!
NEC P6, dt. Handbuch 1145,— NEC CP6, dt. Handb. 1595,—
NEC P2200, dt. Handbuch 969,— Epson LX-800 598,—
NEU: Star LC-10 incl. IBM/Centr. Interf. und D.kabel 629,—

Zubehör für Amiga, Atari und PC sofort lieferbar!

3½″ NN Disk 2DD ab 2,15 bis 2,59 DM
5¼″ NN Disk 2DD ab 0,65 bis 0,85 DM

Über 400 PD Disks lieferbar; 2 Katalogdisks für 8,— DM.
Es sind zur Zeit ca. 200 Spiele und 260 Anwenderprogramme für den Amiga lieferbar. Fragen Sie auch nach Atari- und PC-Hard- und Software. Außerdem eigene Entwicklungen.

C.S.S. Auf der Warte 46, 6367 Karben 1
Telefon 06039/5776



AB-COMPUTERSYSTEME

A. Büdenbender, 5000 Köln 41
Wildenburgstr. 21, ☎ 0221/4301442

IHR Druckerspezialist in Köln
Wir bieten Ihnen noch Beratung und Service für Ihren Computer

NEC P6, 24 Nadeln, komplett, 12 Mon. Garantie mit Treiberdisk	1248,–
NEC P6 Color wie oben, komplett anschlußfertig	1598,–
NEC P2200, 24 Nadeln, Einzelblatt/Endlos, Traktor komplett	998,–
STAR LC 10 NEU, 9 Nadeln, Endlos/Einzelblatt, Traktor	619,–
STAR NB 24/10, 220 Z/Sek. mit Traktor, komplett	1498,–
EPSON LQ 500, 24 Nadeln, kompl. anschlußfertig	998,–
EPSON LQ 850, 24 Nadeln, Super-Ausstattung, komplett	1648,–
GOLEM 2 MB für Amiga 1000/500, komplett	998,–
FILE Card für Amiga 2000, 30 MB, in Verb. PC-Karte	998,–
Disk 3.5 Zoll 2DD, Fuji, eine der besten 10 Stk.	38,–
Disk 3.5 Zoll 2DD, No Name, 100 Stk.	260,–

Software, andere Produkte in unserem Info kostenlos.

Die Preise sind unverbindliche Richtpreise/
Händleranfragen erwünscht/Versand ab 35,– DM

★P★E★G★G★Y★

Bevor Sie sich irgendwelche Programme wie:

- **Copy-Tool**
- **Disk-Monitor**
- **File-Monitor**
- **System-Monitor**
- **CLI-Tool, Shell**
- **Tasten-Editor**
- **Drucker-Spooler**
- **Task-Handler**
- **Fullscreen-Editor**

oder sonst ein Werkzeug zur Programm-Entwicklung anschaffen, sollten Sie sich, **in Ihrem eigenen Interesse,** über *PEGGY* informieren.

PEGGY, die komfortable Programmier-Umgebung von der ersten Zeile bis zum fertigen Programm. Für alle Programmiersprachen geeignet!

PEGGY 2.12 **nur DM 128,–**
PEGGY's Editor incl. Demo nur **DM 10,–**

SAS-BERND
SYSTEM- & ANWENDERSOFTWARE
Langgasse 93 · 5216 Niederkassel 5
☎ (02 28) 45 26 26

AMIGA AUTOKOSTEN

Fuhrpark-Verwaltung

PKW- u.



∗ Für Privat und Firmen.
∗ Gleichzeitige Verwaltung mehrerer Fahrzeuge.
∗ Erfassung und Speicherung aller Ausgaben.
∗ Auch Abschreibungen, Kredite, Treibstoffe.
∗ Wirtschaftliche + statistische Auswertungen.
∗ Kosten- / Leistungsprotokoll.
∗ Leichte Bedienung – alles in Deutsch. DM **98,–**

Hardware-Anforderungen: Amiga 500/1000/2000 mit mind. 1 Disklaufwerk. Centronics-Drucker, 80 Zeichen-Bildschirm

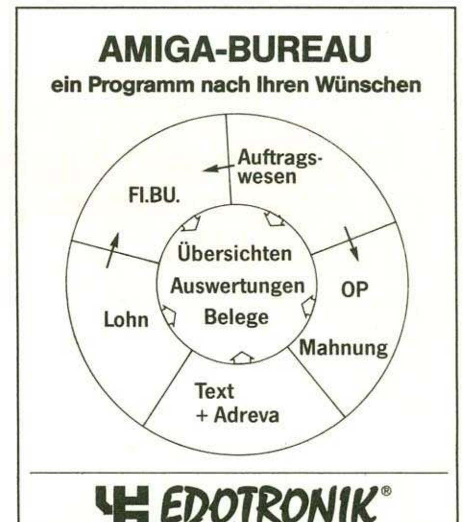
Kostenlosen Sonderprospekt und AMIGA-Katalog anfordern! Besuchen Sie uns im MÜKRA-Ladengeschäft.
Öffnungszeiten: Mo–Fr. 10.00–18.00, Sa. 10.00–13.00 Uhr.
Versand nur per Nachnahme oder Vorkasse (Scheck)
Versandpauschale:
Inland: 6,– / Ausland 12,–

W. Müller & J. Kramke GbR
Schöneberger Str. 5
1000 Berlin 42
Tel. 030-752 91 50/60

mükra
DATEN-TECHNIK

AMIGA-BUREAU

ein Programm nach Ihren Wünschen



EDOTRONIK®

D-8000 München 80, St.-Veit-Straße 70, ☎ 089/40 40 93

AMIGA:

Die Massenspeicher kommen...

20-MB-Festplatte, anschlußfertig
. . . . **DM 1398,–**

40-MB-Festplatte, anschlußfertig
. . . . ab **DM 2498,–**

70-MB-Festplatte, anschlußfertig
. . . . **DM 4198,–**

Weitere Größen auf Anfrage!

P M D 08106/33941
Matthias Aures · Postfach 100105
8011 Vaterstetten

Amiga Speichererweiterung intern, Fastram
1 MB **748,—** 2 MB **1248,—** 4 MB **2048,—**
(Test A-Special 3/87, A-Magazin 12/87)
Laufwerk 1036A, abschaltbar, durchgeschleift, anschlußfertig im amigafarbenen Blechgehäuse (Test Kickstart 10/87, A-Magazin 1/88) **329,—**
Laufwerk 1037A, PROFEX-Drive durchgeführter Bus, Amigafarben, 880 KB, Super-Slimline **299,—**
Laufwerk 5¼ Zoll, abschaltbar, MS-DOS, 40/80 Tracks, durchgeschleift, TEAC **419,—**
CHINON-Laufwerk, 5¼ Zoll, genau wie TEAC, beige Frontblende nur **389,—**
AMIGA 500, 512 K abschaltbar mit Uhr **229,—**
Amiga 500, 2 MB abschaltbar, flaches Gehäuse 512 K od. 2 MB schaltbar (auch f. AMIGA 1000 anschließbar) **919,—**
AMIGA 500, 2 MB extern, durchgeführter Bus, abschaltbar, paßt sich dem 500er Gehäuse an, PROFEX-BOX nur **879,—**
(Test A-Special 5/87, A-Magazin 12/87)
Amiga 1000 Golem-Ram Box **899,—**
Amiga Digitizer A500, 1000, 2000 **99,—**
AMIGA PAL-GENLOCK INTERFACE **545,—**
(reine Hardwaresteuerung per Regler)
(Test A-Special 5/87, A-Magazin 4/88)
FAST-LIGHTNING COPY-PROGRAMM **79,—**
Disketten 2DD original Commodore **29,—**
Coll-Card und Coll-Prom, die ersten Epromkarten und Eprombrenner für den Amiga
Coll-Card A500+A1000 512 K best. **298,—**
weitere 512 K-Karte **49,—** 2 MB **419,—**
Coll-Card A2000 2 MB **398,—**
Coll-Prom A500 + A1000 **398,—**
Coll-Prom A2000 brennt alle 8 Bit Eproms **650,—**
Option bis 16 Bit Eproms **998,—**

Anfragen ab 17.00 Uhr unter 06163/1278 (4326)

F U T U R E V I S I O N
Friedrich-Veith-Str. 21, 6128 Höchst/ODW.

PUBLIC DOMAIN inkl. 2DD Disk 3,–

für Amiga 500, 1000, 2000 z.B. Fish, RW, Faug, SACC, Panorama, ACS, Auge, Amicus ...
Jede 2DD Disk bei Abnahme von:
1- 5 Disks **4,50 DM**
6-10 Disks **3,70 DM**
11-19 Disks **3,50 DM**
ab 20 Disks **3,00 DM**

— alle Disks einzeln nach freier Auswahl
— Disks werden auf modernsten 3½″ Drives dupliziert und auch im »No Name Verfahren« getestet
— alle Disks inkl. Aufkleber (zum Beschriften)
— alle PDs auch auf Markendisks Aufpreis: 1,50 DM
— kein Serienzwang, keine Mindestabnahmemenge

PUBLIC DOMAIN VERZEICHNIS (PDV) auf 3½″ Disks

1 Disk + Infomat. (gesamtes Lieferprogramm) 5,—, 2 Disks + 1 PD Superdisk (Hit des Monats) + ausführliches Infomaterial 10,—. Nur Vorauskasse in bar, keine Schecks !! (Briefm.)

Versand: Nachnahme mit Paketdienst (bei Schwierigkeiten bitte Vermerk:) Post oder Vorauskasse (Bargeld per Einschreiben) Bestellungen von Neukunden bitte nur schriftlich!! (Postkarte genügt!!) Lieferung 24 h wenn lagernd!! Versandkosten **(ca. 9,—/6,—)**, Ausl. **(20,— 15,—)**

Datentechnik M. Bittendorf
Postfach 100248, Abt. Public D., 6360 Friedberg 1,
Hotline Mo-Fr von 9 bis 19 Uhr 06031/61950

AMIGA 500	1098,00
AMIGA 2000	2298,00
AMIGA 2000 mit 1084	2948,00
AMIGA Colormonitor 1084	698,00
Profex CM-14S Color Stereo! - Monitor incl. Kabel für Amiga	648,00
NEC Multisync Monitor	1450,00
Kabel Amiga-NEC Multis.	39,00
Speichererw. 2 MB f. A500	848,00
Amiga Floppy extern	379,00
AMIGA 1010 2. Floppy 3,5	469,00
2000 2. Floppy intern	369,00
2000 PC/AT-Karte	1898,00
Commodore MPS-1200	529,00
Epson LX-800	598,00
Star LC-10 Centronics	648,00
NEC P2200	989,00
NEC P6 bzw. MPS-2000	1198,00
NEC P6 Color/MPS-2000C	1495,00
Amiga Software PC-Emulator für Amiga 1000 KS1.1	99,00
Superbase für Amiga	249,00
Deluxe Paint II	239,00
DATAMAT/TEXTOMAT Amiga	99,00
Micromint AT	2222,00
Multitronik PC-AT	2438,00
Multitronik PC-XT/Monitor	1529,00
Festplatte PC 20 MB	495,00

Electronic-Bauteile/Data Becker, Markt&Technik usw.
Bitte fragen Sie an!
Telefon 07541/73122
Versand und Ladenverkauf
REICH ELECTRONIC
Ailinger Str. 86/1, 7990 Friedrichshafen


NEU!
Chart-Gestaltung und Schriften mit Paint de Luxe im Brush-Modus. Profi-Urteil: Sehr gut !

Chart-Construction-Set. Business-Grafiken im Baukasten-System mit Fertig-Motiven. 2 Disks voller Features. MedRes. 89.-

Headline-Brushes 1. 50 tolle Schriften in LoRes. Für Profis geeignet ! 89.-

Headline-Brushes 2. Viele Profi-Schriften in versch. Schnitten. U.a. Bodoni, Futura, Helvetica, Univers, Avant Garde, LCD, Pump, Eurostyle. Füllmuster. MedRes. 89.-

Digi-Service. Kostenlose Infos anfordern.

Demos zu allen angeboten:
Headline-Brushes 1 und 2 .... je 15.-
Chart-Construction-Set 15.-
Digi-Service 15.-

Alle Preise incl. Versand bei Vorkasse. Nachnahme plus 4.50 Schriftl. Bestellung

Klaus Juris, Computer-Grafik-Design
Bahnhofstraße 106, 6392 Neu-Anspach/Ts.

Rainbow Data

Speichererweiterung für

Amiga 500
512-KB-RAM-Speicherkapazität, m. Uhr u. Abschaltung ca. **239,–**

Druckerkabel Amiga 500/2000 **23,—**

Druckerkabel Amiga 1000 **23,—**

Monitorkabel Amiga/Scart **29,—**

Rainbow Data
Am Kalkofen 1, 5603 Wülfrath
Telefon 02058/1366

Wir verkaufen Qualität . . . Preiswert .

Codat EDV-Service
Gerd Drüppel
Roonstr. 28 4690 Herne 1
nur Versand
02323 / 83564

AMIGA *** AMIGA

Spiele Utilities Grafik Musik Anleitungen

Fish
Faug
Amicus
Tbag
Auge
Panorama
Amigacasa
Amuse
und mehr

1000'er 500'er 2000'er

Katalogdisk gegen DM 5.-
Dazu gibts ein Spiel ! gratis !

Versand der Public Domain Software nur gegen: Vorkasse (Scheck) bzw. Nachnahme

Chris Schumacher
Wormser Str. 9
5000 Koeln 1
Tel. 0221/372391

1 - 10 St. DM 8.-
11 - 50 St. DM 6.-
ab 51 St. DM 5.-
alle Preise inc. MwSt. und 3,5 " 2 DD - Disk.

Public Domain

AMIGA - SOFTWARE

Public Domain Disketten

Greifen Sie **jetzt** zu ! **Superangebote !**

NEU
10 Disketten Ihrer Wahl nur **48DM**
20 Disketten Ihrer Wahl nur **96DM**
30 Disketten Ihrer Wahl nur **144DM**
obige Preise incl. Versandkosten

Wählen Sie aus Fish, Faug, Panorama, TBAG, AUGE, Amicus, Chiron Conception, u. v. a.

NEU
Professionelles elektronic Worksheet incl. Source auf 3 Disketten **24 DM**

Einzel- / Katalog Diskette gegen 6,40 DM als Scheck oder Briefmarken incl. Versandkosten.
- Disketten 2DD Commodore **39,75 DM**
- Zweitlaufwerk Amiga 500/1000 **328DM**

Sie erreichen uns auch nach 18 Uhr.

A. Fischer, Kirchstr. 40, Tel. 05257- 4347
4794 Hövelhof

FASTWORKS

PUBLIC DOMAIN inkl. 2DD-DISK4,– DM!
FISH 1-135, PANORAMA 1-62, FAUG 1-39, AUGE 1-12, AMICUS 1-20 und natürlich die Originalen: TORNADOS Nr. 1-30 (6 DM je Disk), Katalogdisk PD und TORNADOS je 5 DM
BOOTWRITER 17 DM - IFF-CON 27 DM
Porto/Verpackung: Inland = 4 DM;
Ausland = 7 DM je Bestellung

Baller-Games sind ja ganz schön..
Aber jetzt gibt es doch:

M·A·N·A·G·E·R

Das WIRTSCHAFTSSPIEL für bis zu 20 Spieler!
Firma gründen ist nicht schwer, doch oben bleiben.
Unv. Preisempf. 49,50 DM zzgl. 5 DM (Ausl. 9 DM Porto/Verp.)

ACHTUNG: **PROGRAMMIERER** gesucht.
Garantiert bis 50% Umsatzbeteiligung.

Bestellungen bar oder V-Scheck (Ausl. nur Vorkasse)

FASTWORKS, Fichtestr. 16, 5090 Leverkusen 1
Telefonische Bestellannahme unter 0214/94668

AMIGA-SOFTWARE ZU SUPERPREISEN

incl. 2 DD Diskette ab 3.65 DM

Fred-Fish .... **Nr. 1-135**	FAUG .... **Nr. 1-51**
PANORMA .. **Nr. 1- 62**	AUGE 4000 ... **Nr. 1-14**
Amicus ...... **Nr. 1- 20**	und andere ....

(Katalogdisk gegen 5.- DM anfordern)

Einzeldisk . **4.70 je Disk**	ab 40 St. .. **4.20 je Disk**
ab 10 St. ... **4.50 je Disk**	ab 100 St. . **3.90 je Disk**
ab 20 St. ... **4.40 je Disk**	274 St. nach Wahl **999.00**

alle Preise incl. 2 DD Diskette

TORNADOS - Die Super PD-Serie. Siehe AMIGA 01/88 u. 03/88. Katalogdisk gegen 5.00 anfordern.
Nr. 1-30 (incl. 2 DD Disk) .... **4.80 je Disk**
BOOTWRITER V 1.0 - kostenl. Infos **17.00**
IFF-CON 0.9 - kostenl. Infos **27.00**

NEU - Versenden Sie Ihre Disketten in einer „bombensicheren" Verpackung ....
Speziell gepolsterte Versandtaschen: Preis in DM per 100 Stück

	ab 100	ab 500	ab 1000
Innenmaß mm 127 x 229:	47.00	44.00	42.00
Innenmaß mm 152 x 229:	54.00	51.00	49.00
Innenmaß mm 184 x 280:	67.00	62.00	59.00

Inland: Porto/Verp. = 3.00 - Ausland: Porto/Verp. = 6.00
Lieferung gegen Vorkasse oder V-Scheck
Achtung! Kein Ladenverkauf!

Bestellung und Anfragen an:
PD-Shop
Opladener Straße 30, D-4018 Langenfeld

Laufwerke für Amiga

Externes Laufwerk in Metallgehäuse, durchgeschliffener BUS, Laufwerk um-/abschaltbar.
3½", 1 MB, 2x80 Sp. 290 DM
5¼", 1 MB, 2x80/40 Sp. 375 DM
Speichererweiterung 512 KB für Amiga 500 mit Uhr und Kalender 275 DM

Angelika Huber
Elektr. Bedarf
Wörnitzstraße 3, 8850 Donauwörth
Telefon 0906/5567

Rechtschreibung einfach leichtgemacht!

Deutsche Sprache, schwere Sprache. Beim schnellen Eintippen von Texten macht man oft Fehler. »EasySpelling« hilft Ihnen, Ihren Text zu optimieren. Dabei ist die Bedienung des sehr kurzen Programms kinderleicht.

Jetzt können Sie den Kampf mit der deutschen Rechtschreibung endlich aufnehmen, da Ihnen »EasySpelling« zur Seite steht. Das Programm überprüft Ihren gesamten Text auf Groß- und Kleinschreibung, Umlaute sowie doppelte Buchstaben. Auch wenn das Programm (siehe Listing) sehr kurz erscheint, erfüllt es seinen Zweck zur vollsten Zufriedenheit. Jedenfalls bestand es alle Tests, denen wir es unterworfen haben. EasySpelling ist vor allem gedacht für Leute, die sehr viel schreiben müssen.

di bedinung des programs ist denkbar einfak. surst rufen si easispeling (ein wirklik trefender name) fom kli auf:

```
EasySpelling
```

Es erscheint die Startmeldung. Nun erwartet das Programm die Eingabe eines Dateinamens. In dieser Datei muß Ihr unkorrigierter Text stehen. Hier können Sie natürlich einen kompletten Pfadnamen übergeben. Ein Beispiel ist:

```
DF1:Texte/Versuch.txt
```

Der Quelltext muß im ASCII-Format vorliegen, in dem man aber praktisch von jedem Textverarbeitungsprogramm aus speichern kann. Programmeditoren wie ED verwenden diesen Modus sowieso. Kann EasySpelling die angegebene Datei nicht finden, schließt es das Fenster und gibt eine Fehlermeldung im CLI-Fenster aus. Nun müssen Sie den Namen der Zieldatei eintippen.

ä	a	ae	a	ch	k
ö	o	ue	u	ck	k
ü	u	oe	o	c	k
ah	a	ß	s	j	i
eh	e	v	f	y	i
ih	i	ph	f	ts	s
oh	o	z	s	th	t
uh	u	sch	s	pf	f
ie	i	q	k		

Das ist die Konvertierungstabelle von »EasySpelling«.

Auch hier sind Pfadnamen erlaubt. Als zusätzliche Ausgabegeräte stehen noch »prt:« (Drucker) und »con:« (Bildschirm) zur Verfügung. Wegen der möglichen Druckerbenutzung muß auf Ihrer Workbench-Diskette der richtige Druckertreiber im Verzeichnis »:devs/printers« zur Verfügung stehen.

si konen den text also als kontrole auf dem bildsirm ausgeben lasen. wen si den text nur einmal ausgeben wolen, emfilt es sik in gleik druken su lasen.

Wird nicht »con:« verwendet, zeigt EasySpelling die Nummer der gerade bearbeiteten Zeile an. Somit sehen Sie ständig den Fortschritt der Arbeit. Das Listing selbst ist leicht verständlich und kann ohne Probleme erweitert werden. Wie wäre es zum Beispiel mit einer Option zur Unterdrückung der Satzzeichen?

nak der uberarbeitung des textes mit easispeling ist ir srifstuk iedenfals deutlik einfaker su lesen. und di meisten komputerfreaks sind ia fon fereinfakungen ieder art begeistert.

EasySpelling ist ein gutes Beispiel für Konvertierungsprogramme, die man gut an persönliche Bedürfnisse anpassen kann. Im Augenblick gibt es 26 Suchbegriffe, die das Programm ersetzt. Danach folgen die zu ersetzenden Zeichen und die Ersatzzeichen.

Die notwendigen Aufrufe für den Aztec-C-Compiler und Linker sehen so aus:

```
cc EasySpelling +L -S
ln EasySpelling.o -LC32
```

Die Zeilen, in denen »rewind(console);« steht, müssen Sie beim Aztec-C-Compiler nicht eingeben.

Als Lattice-Benutzer verwenden Sie die folgenden Befehle:

```
lc EasySpelling
blink lib:c.o,es.o lib lib:lc.lib,lib:amiga.lib
```

Die Warnungen können dabei ignoriert werden.

wolen si anders konfertiren, so ersesen si di notwendigen seiken und sesen »ansal« auf den riktigen wert. danak funktionirt easispeling fur iren personliken anwendungsfal.

iedenfals erhalten si mit easispeling ein ser leistungsfaiges program, das inen bei der swirigen deusen reksreibung hilft. auk di fon inen angesribenen werden sik freun, da di brife in sukunft kurser und leikter lesbar sind. das spart seit und seit ist bekantlik geld.

(Stefan Tandecki/rb)

Da dieses Programm sicherlich sehr viele Leser interessiert, wäre es ganz toll, wenn ein Leser, der beide Sprachen (Basic und C) beherrscht, dieses Listing umsetzt. Wir werden es dann ganz sicher auf die Programmservice-Diskette geben. Dadurch kann es dann jeder benutzen.

Programmname:	EasySpelling
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	C
Compiler:	Aztec-C V3.4 oder Lattice-C V3.1
Aufrufe:	siehe Text

```
Programm : EasySpelling
-----------------------

v80 /* EasySpelling.C  1988 von Stefan Tandecki */
BM  #include <stdio.h>
60  #include <exec/types.h>
pu  #define ANZAHL 26
cz  char *FindWhat[ANZAHL] = {"ä","ö","ü","ah","eh","ih","oh","u
    h","ie","ae","ue",
SvQ                             "oe","ß","v","ph","z","sch","q","c
                                h","ck","c","j",
2B                              "y","ts","th","pf"},
Cj5      *ReplaceWith[ANZAHL] = {"a","o","u","a","e","i","o","u"
         ,"i","a","u","o",
4XT                                "s","f","f","s","s","k","k","k"
                                   ,"k","i","i","s",
7h                                 "t","f"};
uu0 char quelle[50],ziel[50],zeile[100];
El  short len;
EN  FILE *ein,*aus,*console,*printer,*fopen();
oo  /* pos - Durchsuchen der Zeichenkette Where nach der Zeichen
    kette What. */
MH  short pos(What,Where)
```

Listing. »EasySpelling« ist eine fantastische Hilfe für Rechtschreibprobleme.
Bitte mit dem Checksummer eingeben.

```
char *What,*Where;
{
 short p1,p2;
 p1 = p2 = 0;
 do
 {
  if (*(What + p1) == *(Where + p2)) p1++; else p1 = 0;
  p2++;
 } while (( p1 < strlen(What)) && (p2 < len));
 if (p1 != strlen(What)) p2 = 0; else p2 = p2-strlen(What)+1;
 return(p2);
}
/* GetLine - Einlesen einer Zeile (bis (char)10 oder EOF) aus der Datei */
BOOL GetLine(From,To)
FILE *From;
char *To;
{
 int c;
 short lastchar;
 len = lastchar = -1;
 do
 {
  len++; if ((c = fgetc(From)) == EOF) return(FALSE);
  if (c != 10)
  {
   *(To + len) = c;
   if (c != 32) lastchar = len;
  }
 } while (c != 10);
 *(To + lastchar + 1) = 0; len = lastchar + 1;
 return(TRUE);
}
/* WaitForKey - Warten, bis die Return-Taste gedrückt wird. */
WaitForKey()
{
 char inp[2],*pointer;
 do
 {
  rewind(console);
  pointer = fgets(inp,2,console);
 } while (*pointer != (char)10);
}
/*
    SetUp - Konsole und Druckerkanal öffnen.
 */
SetUp()
{
 console = fopen("CON:0/0/640/200/EasySpelling","w+");
 if (!(console))
 {
  printf("Konsole kann nicht geöffnet werden !!\n"); exit(FALSE);
 }
 printer = fopen("PRT:","w");
 if (!(printer))
 {
  fclose(console);
  printf("Druckerkanal kann nicht geöffnet werden !!\n"); exit(FALSE);
 }
}
/* InputFiles - Quell- und Zieldatei abfragen. */
InputFiles()
{
 rewind(console);
 fprintf(console,"\nQuelldatei: ");
 rewind(console);
 fscanf(console,"%50s",&quelle);
 rewind(console);
 ein = fopen(&quelle,"r");
 if (!(ein))
 {
  fclose(printer); fclose(console);
  printf("%s konnte nicht geöffnet werden !!\n",&quelle); exit(FALSE);
 }
 fprintf(console,"\nZieldatei (con: = Bildschirm, prt: = Drucker): ");
 rewind(console);
 fscanf(console,"%50s",&ziel);
 rewind(console);
 if (!(strcmp(&ziel,"con:"))) aus = console;
  else if (!(strcmp(&ziel,"prt:"))) aus = printer;
  else aus = fopen(&ziel,"w");
 if (!(aus))
 {
  fclose(printer); fclose(console);
  printf("%s konnte nicht geöffnet werden !!\n",&ziel); exit(FALSE);
 }
}
/* Convert - Vereinfachungsroutine. */
Convert()
{
 char temp[100],a;
 short i,e,p,rl,fl;
 for (i=0; i<len; i++)
 {
  a = zeile[i];
  if ((a >= 'A') && (a <= 'Z')) a += 32;
   else if (a == 'Ä') a = 'ä';
   else if (a == 'Ö') a = 'ö';
   else if (a == 'Ü') a = 'ü';
  zeile[i] = a;
 }
 i = 0;
 do  /* Eliminierung der unnützen Buchstabenverdopplung. */
 {
  if ((zeile[i] == zeile[i+1]) && (zeile[i] > 'A') && (zeile[i] < 'z'))
  {
   for (e=i; e<len; e++) zeile[e] = zeile[e+1];
   len--;
  }
  else i++;
 } while (i < len);
 for (i=0; i<ANZAHL; i++)
  do
  {
   p = pos(FindWhat[i],&zeile);
   if (p)
   {
    fl = strlen(FindWhat[i]); rl = strlen(ReplaceWith[i]);
    movmem(&zeile[p+fl-1],&temp,len-p);
    movmem(ReplaceWith[i],&zeile[p-1],rl);
    movmem(&temp,&zeile[p+rl-1],len-p);
    len = len + rl - fl;
   }
  } while (p);
}
/*
    CleanUp - Alle Dateien schließen.
 */
CleanUp()
{
 if (aus != printer) fclose(printer);
 if (aus == console) WaitForKey(); else fclose(console);
 fclose(aus); fclose(ein); exit(TRUE);
}
main()
{
 int line;
 SetUp();
 fprintf(console,"\f\n\nE A S Y S P E L L I N G\n");
 fprintf(console,"----------------------------------\n\n");
 fprintf(console,"Programm zum Vereinfachen der Schreibweise eines Textes.\n");
 fprintf(console,"Umgesetzt 1988 von Stefan Tandecki, 3307 Schöppenstedt.\n");
 InputFiles();
 line = 1; fprintf(console,"\n\n");
 while (GetLine(ein,&zeile))
 {
  if (len > 0) Convert();
  fprintf(aus,"%s\n",&zeile);
  if (aus != console) fprintf(console,"\rZeile: %5ld",line);
  line++;
 }
 CleanUp();
}
```


Listing. »EasySpelling« ist eine fantastische Hilfe für Rechtschreibprobleme. Bitte mit dem Checksummer eingeben.

☆ ☆ **AMIGA** ☆ ☆

SPRACHEN/ENTWICKLUNG

Metacomco Assembler	159,--
Metacomco Pascal	239,--
Lattice C-Compiler	399,--

SPIELE

Bad Cat	54,--
Barbarian (Psygnosis)	66,--
Bard's Tale	89,--
Bureaucracy	89,--
Chessmaster 2000	82,--
Dark Castle	69,--
Defender of the Crown	79,--
Deja Vu	79,--
Flight Simulator II	119,--
Garrison II	64,--
Goldrunner	69,--
Impact	44,--
Jinxter	72,--
King of Chicago	64,--
Kings Quest III	76,--
Mission Elevator	59,--
Shadowgate	69,--
Terrorpods	66,--
Test Drive	79,--
The Guild of Thieves	69,--
The Feary Tale	119,--
The Pawn	72,--
Ultima III	72,--
Ultima IV	72,--
Uninvited	79,--
Western Games	59,--

DRUCKER

NEC P 2200	1049,--
Epson LQ-500	998,--
Epson LX-800	699,--
Star LC10	649,--

☞ Sofort kostenlos Preisliste bei Abteilung AM anfordern! ☜

Computer & Zubehör Versand Gerhard und Bernd Waller GbR

Kieler Str. 623, 2000 Hamburg 54, ☎ 040/5706007 + 5705275

Genlock DM 498,–



ImaGen!



- High Tech Konstruktion auf wenigen Chips!
- Super leistungsfähige Hardware und Software
- Deutsches Handbuch, deutsche Pal-Version
- Greifen Sie zu, reservieren Sie sich Genlock zum Superpreis!
- Leistungsfähiger als manche Geräte für über 1000 DM
- In USA über 10.000 mal in kürzester Zeit verkauft!

IM Intelligent Memory

Borsigallee 18
6000 Frankfurt/M.
☎ 069/410071/72

Schweiz:
MICROTRON
Bahnhofstraße 2
CH-2542 Pieterlen
Tel 032 87 24 29

NEU FÜR AMIGA:

Druck-Master

Druckt IFF - Grafiken in nahezu Fotoqualität (siehe Demo-Bilder).

- **Formate von DIN A 6 - DIN A 2 möglich**
- **Läuft auf allen AMIGA-Modellen in Verbindung mit NEC P6 / P7 oder kompatiblen Druckern.**



AM BESTEN GLEICH BESTELLEN !

Druck-Master **Best.-Nr.: A - 01 001 88**

●●●●● **89. – DM** ●●●●●

Gegen 1,30 DM in Briefmarken erhalten Sie unsere **Info-Blätter** über unser derzeitiges Angebot an AMIGA-Software.

Bestellungen unter:
Lange Straße 51, 2320 Plön
Telefon: 04522 / 1379

SOFTWARE 2000

DISKETTENLAUFWERKE

vollkompatibel, anschlußfertig, inkl. Kabel, Busbox, Metallgehäuse, 2 x 80 Tracks, 1 MB unform., 3 ms Steprate, 25,4 mm flach, automatische Diskchange-Erkennung, abschlatbar, preisgünstige Anschlußmöglichkeit für ein weiteres Laufwerk. 3,5": **229.-** 5,25": **259.-**

➡ 1. Floppybussystem ⬅

3,5" Qualitätslaufwerke, 25,4 mm flach, modernste Technik, anschlußfertig und vollkompatibel.

NEC 1037 oder
TEAC FD 135
inkl. Busbox

298.-

Diskettenlaufwerke:

NEC FD 1037	**199.-**
TEAC FD 55 FR	**229.-**
TEAC FD 135 FN	**209.-**

Anschlußfertig für Busbox:

NEC oder TEAC 3,5":	**229.-**
TEAC 5,25"	**259.-**

Busbox **98.-**

Ermöglicht den Anschluß von 2 beliebigen Shugart-Bus (alle handelsüblichen) Floppylaufwerken. Alle angeschlossenen Laufwerke können einzeln abgeschaltet werden. Driveselect schaltbar.

➡ TEAC 5.25" ⬅

FD 55 FR, 1 MB, 2 x 80 Tracks, anschlußfertig und vollkompatibel

40/80 Tracks schaltbar,
MsDos kompatibel,
inkl. Busbox

359.-

Weitere Informationen erhalten Sie bei:
Postleitzahlengebiete: 1, 6, 7, 8: **FSE - Frank Strauß Elektronik**
St. Marienplatz 7 · 6750 Kaiserslautern · Tel. 0631 / 16258
Postleitzahlengebiete: 2, 3, 4, 5: **Max Melchior GmbH**
Kleine Straße 3 · 5650 Solingen · Tel. 0212 / 50773

AMIGA HARD- und SOFTWARE

ERAM 500

512 KB-RAM für den Amiga 500 • gepufferte Uhr • abschaltbar + 5 DM •

Preis a. A.

EXPO I

Experimentierkarte für den Amiga 2000 • Mit Adreßauswahl für 8 ICs • I/O-Baustein • Adreß-, Daten, FC-Signale •

199,-

USPA II

Kickstart Umschaltplatine • Ihre eigene Version auf EPROMs • Platine o. EPROMs

59,-

Externes Laufwerk 3½", abschaltbar, mit durchgeführtem Bus **348,-**

Großes SOFTWARE-Angebot! Fordern Sie unseren Katalog an, telefonisch oder gegen Rückumschlag.

Telefon 02232/13063 + 47105

Ralf Tröps ⋆ Computertechnik ⋆ 5040 Brühl ⋆ Pingsdorferstr. 141



Gefährliche Sportspiele

Das Spiel mit den zwei Motorrädern, die sich den Weg abschneiden, ist wohl sehr beliebt. Aus diesem Grund hier eine sehr gute Basic-Version, damit auch wirklich jeder Amiga-Besitzer spielen kann.

Troof ist eine Sportart, die im Jahr 2599 erfunden wurde. Der Grund dafür war das mangelnde Interesse an den alten Wettkämpfen. Die Grundidee ist schnell erklärt: Mit abgewandelten Motorrädern fahren die zwei Spieler über die Fläche. Dabei dürfen Sie weder Wände noch die Spur des Gegners berühren. Für die Fahrer unsichtbar läuft eine Uhr von 1000 rückwärts. Muß nun einer der Kontrahenten sein Fahrzeug mit dem Schleudersitz verlassen, wird dem anderen der auf der Uhr angezeigte Betrag in Dollar gutgeschrieben. Nach Beendigung einer vorgegebenen Zahl von Runden wird dann der Sieger begeistert vom Publikum gefeiert.

Da wir aber noch nicht das Jahr 2599 schreiben, folgt jetzt die Bedienungsanleitung für Troof auf dem Amiga.

Nach einem Doppelklick auf das Symbol des Programms und einer kurzen Wartezeit startet Troof. Auf dem Bildschirm erscheint ein Schriftzug und die Aufforderung, eine der zwei möglichen Tasten auszuwählen.

Wenn Sie sofort auf <SPACE> drücken, haben die zwei Spieler keinen Namen und ein weiterer Bildschirm erscheint.

Wollen Sie jedoch Namen vergeben, so drücken Sie <F1>. Auf dem erscheinenden Bild geben Sie zunächst die Namen der zwei Konkurrenten, jeweils gefolgt von <RETURN>, ein. Nun können Sie die Anzahl der zu spielenden Runden wählen.

```
F1      5 Runden
F2      9
F3      13
F4      17
F5      19
```

Sollten Sie einen Fehler begangen haben, können Sie die Eingaben wiederholen, wenn Sie <Esc> drücken. Ist alles zu Ihrer Zufriedenheit ausgefüllt, starten Sie das Spiel mit <SPACE>.

Es erscheint nun die Anzeige des aktuellen Punktestands. Wollen Sie Motorengeräusche beim Spielen, drücken Sie jetzt <F1>. Dadurch läuft das Spiel etwas langsamer, was am Anfang meist nicht von Übel ist. Drücken Sie nun <SPACE>.

Nun zur Steuerung der Motorräder. Die Richtung kann nur um 90 Grad nach rechts oder links verändert werden. Die notwendigen Tasten für Spieler 1 und 2 sehen Sie in der Tabelle. Achten Sie dabei darauf, daß <Caps Lock> nicht eingeschaltet ist. Die Zahlen zum Steuern sind die des Ziffernblocks rechts auf der Tastatur.

Die ersten Spiele sind besonders interessant, da man noch nicht genau weiß, was man berühren darf und was nicht. Ein kleiner Tip: Es geht um verschiedene Farben. Aber auch später verliert das Spiel kaum an Reiz, da die verschiedenen Hintergründe keine Langeweile aufkommen lassen. Auf besonderen Wunsch des Autors sei noch erwähnt, daß die Steuerung von »Hille« programmiert ist.

(Patrick Kälin/rb)

	Links	Rechts
Spieler 1	a	d
Spieler 2	4	6

Tabelle. Die Tasten für die Richtungsänderungen

Programmname:	Troof
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2

```
Programm : Troof
----------------

1D1  CLEAR,25999
C3   DIM x1(20):DIM x2(20):DIM y1(20):DIM y2(2)
12   SCREEN 1,320,200,5,1
Y4   WINDOW 2,"TROOF PROGRAMMED 1987 BY PAT", ,0,1
Z70 Start:
fi1  PALETTE 0,0,0,0
2y   PALETTE 1,.5,.45,.2
36   PALETTE 2,1,1,1
Ci   PALETTE 3,1,.73,0
3A   PALETTE 4,0,0,0
r4   PALETTE 5,.83,.2,0
3q   PALETTE 6,0,.6,0
A6   PALETTE 7,.8,.6,.33
6r   PALETTE 8,.15,0,0
t9   PALETTE 9,.8,.8,0
fZ   PALETTE 10,1,1,.13
HH   PALETTE 11,.5,.5,.4
o1   PALETTE 12,0,0,1
tL   PALETTE 13,0,0,.6
hW   PALETTE 14,.5,.5,.5
S7   PALETTE 15,.3,.3,.3
H4   PALETTE 16,.7,.7,.7
V5   PALETTE 17,0,.3,0
Ib   PALETTE 18,0,.93,.87
DL   PALETTE 19,0,.8,0
qC   PALETTE 20,1,0,0
VL   PALETTE 21,.8,.6,.33
ma   PALETTE 22,.8,.6,.53
XM   PALETTE 23,.4,.3,.3
aQ   PALETTE 24,0,.4,0 :REM gruen
BI   PALETTE 25,.8,.9,1:REM
gd   PALETTE 26,.6,.9,1 :REM
o6   PALETTE 27,.9,.9,.9:REM
Ds   PALETTE 28,.1,.1,.1: REM sehr grau
s0   REM
1b   REM titel
u2   REM
2Y   titel:
BH   CLS
m8   FOR a=30 TO 50 STEP 3
Oy   LINE(5,a)-(65,a),5
xI   LINE(75,a)-(125,a),5
3f   LINE(135,a)-(185,a),5
32   LINE(195,a)-(245,a),5
bo   LINE(255,a)-(305,a),5
pu   NEXT
Fh   FOR a=52 TO 70 STEP 3
fA   LINE(25,a)-(45,a),9
SX   LINE(75,a)-(125,a),9
Iy   LINE(135,a)-(185,a),9
IL   LINE(195,a)-(245,a),9
f7   LINE(295,70)-(255,a),9
w1   NEXT
a6   FOR a=72 TO 90 STEP 3
YJ   LINE(25,a)-(45,a),19
3y   LINE(75,a)-(95,a),19
Uv   LINE(115,90)-(105,a),19
NK   LINE(135,a)-(185,a),19
Nh   LINE(195,a)-(245,a),19
Wb   LINE(255,a)-(275,a),19
49   NEXT
fr   FOR a=92 TO 105 STEP 3
BL   LINE(25,a)-(46,a),13
bu   LINE(75,a)-(95,a),13
4W   LINE(125,104)-(117,a),13
Sk   LINE(105,a)-(117,a),13
E5   LINE(135,a)-(185,a),13
ES   LINE(195,a)-(245,a),13
NM   LINE(255,a)-(275,a),13
DI   NEXT
F2   FOR a=6 TO 310 STEP 3
zX   LINE(a,30)-(a,105),4
GL   NEXT
QX0 titlmsk:
3U1  RESTORE
7J   FOR a=1 TO 18
9v   READ ton:READ dauer
gw   SOUND ton,dauer,50,0
MR   NEXT
```

```
iO   DATA 208,2,220,4,261,4
wC   DATA 261,3,261,2,261,4
jX   DATA 329,3,293 ,2,293,3
85   DATA 261,2,261,3,246,2
OD   DATA 261,4,220,3,196,3
OM   DATA 174,3,164,4,220,3
Qr   FOR w=200 TO 100 STEP -10
24   SOUND w,1,70,0
Va   NEXT
k50  key0:
Y61  a$=INKEY$
VS   COLOR 8
JA   LOCATE 2,2:PRINT"BY PATRICK KAELIN IN 1987"
a0   LOCATE 14,5:PRINT"THANKS TO KUSI & HILLE"
8y   LOCATE 16,3:PRINT"SPACE = BEGINN THE GAME"
te   FOR a=1 TO 300:NEXT
hg   LOCATE 18,3:PRINT"  F1  = SELECT THE NAME"
SM   COLOR 5
PG   LOCATE 2,2:PRINT"BY PATRICK KAELIN IN 1987"
ux   LOCATE 14,5:   PRINT "THANKS TO KUSI & HILLE"
06   LOCATE 16,3:PRINT"SPACE = BEGIN  THE GAME"
ml   LOCATE 18,3:PRINT"  F1  = SELECT THE NAME"
8f   IF a$=" " THEN BC
mF   IF a$=CHR$(129) THEN GOTO namen
99   GOTO key0
HK0  namen:
GM1  CLS
vL   GOSUB rahmen:COLOR 9
qJ   LOCATE 4,6:PRINT "SPIELER 1 :RED VEHICLE "
jY   LOCATE 6,6:PRINT "SPIELER 2 :GREEN VEHICLE":COLOR 1
Ni   LOCATE 9,6:PRINT"PRESS F1/F2/F3/F4/F5"
BX   LOCATE 11,2:PRINT"FUER 5 9 13 17 ODER 19 RUNDEN"
s0   LOCATE 13,6:PRINT"ESC FUER NEUE CHANCE":COLOR 9
42   LOCATE 17,6:INPUT"SPIELERNAME 1";s$
Z8   IF LEN(s$)>10 THEN namen
Cr   LOCATE 19,6:INPUT"SPIELERNAME 2";S2$
kg   IF LEN(S2$)>10 THEN namen
Hd0  key1:
OY1  a$=INKEY$
ye   IF a$=CHR$(129) THEN runden=6
Tu   IF a$=CHR$(130) THEN runden=10
x9   IF a$=CHR$(131) THEN runden=14
RO   IF a$=CHR$(132) THEN runden=18
qL   IF a$=CHR$(133) THEN runden=20
tY   IF a$=CHR$(27) THEN GOTO namen
V6   IF a$=" "THEN BC
ab   GOTO key1
W30  BC:
uH1  FOR a=0 TO 320
1y   LINE(a,0)-(a,200),4:NEXT
cY   le=0:P=0:P2=0:wi=0
Vk0  punkte:
e61  x1=150:x2=170:y1=100:y2=100
hv   le=le+1:b1=1:b2=1
io   CLS
DX   ti=1000:dauer=0
Oo   GOSUB rahmen
hc   LOCATE 4,4:COLOR 1:PRINT "PLAYER"wi;:PRINT "IST DER GEWINNER":COLOR 9
s4   LOCATE 8,3 :PRINT"# 1 : ";:PRINT s$
BA   LOCATE 8,17:PRINT" = ";:PRINT P;:PRINT "$"
Li   LOCATE 10,3 :PRINT"# 2 : ";:PRINT S2$
wr   LOCATE 10,17:PRINT" = ";:PRINT P2;:PRINT "$"
2t   LOCATE 14,11:COLOR 1:PRINT"GOOD LUCK !"
Db   LOCATE 18,7:COLOR 9:PRINT"PRESS F1 FOR SOUND !"
nA0  key2:
Rz1  a$=INKEY$
PI   IF a$=" " THEN ax
n4   IF a$=CHR$(129) THEN dauer=.1
13   GOTO key2
RK0  ax:
v01  x(1)=0:x(2)=1:x(3)=0:x(4)=-1
1A   y(1)=-1:y(2)=0:y(3)=1:y(4)=0
0Y   CLS:GOSUB feld
1P0  key3:
1I1  SOUND 150,dauer,70,0
JM   PSET(x1,y1),5:PSET(x2,y2),6
cA   a$=INKEY$
G8   x1=x1+x(b1):y1=y1+y(b1)
DK   IF ti=0 THEN ti=1000
QB   IF a$="a" THEN b1=b1-1:IF b1=0 THEN b1=4
1P   IF a$="d" THEN b1=b1+1:IF b1=5 THEN b1=1
nv   ti=ti-1
1Z   x2=x2+x(b2):y2=y2+y(b2)
rs   IF a$="4" THEN b2=b2-1:IF b2=0 THEN b2=4
81   IF a$="6" THEN b2=b2+1:IF b2=5 THEN b2=1
qL   IF POINT(x1,y1) = 5 OR POINT(x1,y1)=6 OR POINT (x1,y1)=15 THEN GOTO crash
Qt   IF POINT(x2,y2) = 5 OR POINT(x2,y2)=6 OR POINT(x2,y2)=15 THEN GOTO crash2
PS   GOTO key3
X60  crash:
Qc1  FOR m=1 TO 2:FOR a=2 TO 9
sB2   CIRCLE(x1,y1),a,2:CIRCLE(x1,y1),(a+2),9
bA1  FOR s=80 TO 120 STEP 10
1p   SOUND s,.1,250,0:NEXT
vV   CIRCLE(x1,y1),(a+4),22:CIRCLE(x1,y1),(a+6),15
mL   CIRCLE(x1,y1),(a+9),16
Vz   NEXT:NEXT
nl   FOR s=18 TO 0 STEP -1
eE   SOUND s*10,.1,90,0
iK   CIRCLE(x1,y1),s,4
cy   NEXT:wi=2:P2=P2+1*ti:punkt=P2:w$=S2$:GOTO punkte
ja0  crash2:
co1  FOR m=1 TO 2:FOR a=2 TO 9
Ld   CIRCLE(x2,y2),a,2:CIRCLE(x2,y2),(a+2),9
4Y   FOR s= 80 TO 120 STEP 10
x1   SOUND s,.1,250,0:NEXT
Ox   CIRCLE(x2,y2),(a+4),22:CIRCLE(x2,y2),(a+6),15
gJ   CIRCLE(x2,y2),(a+9),16:NEXT:NEXT
yw   FOR s=18 TO 0 STEP -1
pP   SOUND s*10,.1,90,0
1f   CIRCLE(x2,y2),s,4
nm   NEXT:wi=1:P=P+1+ti:punkt=P:w$=s$:GOTO punkte
aT0  feld:
NC1  IF runden=le GOTO sieger
Wg   IF le=1 THEN level1
fr   IF le=2 THEN level2
o2   IF le=3 THEN level3
xD   IF le=4 THEN level4
60   IF le=5 THEN level5
FZ   IF le=6 THEN level6
Ok   IF le=7 THEN level7
Xv   IF le=8 THEN level8
g6   IF le=9 THEN level9
mg   IF le=10  THEN  level10
vr   IF le=11  THEN  level11
42   IF le=12  THEN  level12
DD   IF le=13  THEN  level13
MO   IF le=14  THEN  level14
VZ   IF le=15  THEN  level15
ek   IF le=16  THEN  level16
nv   IF le=17  THEN  level17
w6   IF le=18  THEN  level18
bf   IF le=19 THEN level19
tc   IF le=20 THEN level20
01   IF le=21 THEN level21
7a   IF le=22 THEN GOTO sieger
D8   END
SW0  sieger:
J7   GOSUB rahmen
D3   COLOR 1
Dw   LOCATE 3,8:PRINT" GESAMMT-BILANZ"
1h   LINE(70,30)-(220,30),9
ZK   LINE(70,31)-(220,31),1
gl   COLOR 7:LOCATE 6,4:PRINT"# 1 : "s$:
1K   LOCATE 6,19:PRINT" = "P
tx   LOCATE 8,4:PRINT"# 2 : "S2$
NG   LOCATE 8,19:PRINT" = "P2
6t   IF P>P2 THEN w$=s$
bI   IF P<P2 THEN w$=S2$
ND   COLOR 1
BW   LOCATE 14,5:PRINT"THE WINNER : "w$
Td   r1=200:r2=500:REM fuer sound
nf   KEY5:
q01  a$=INKEY$
mJ   IF a$=" " THEN GOTO Start
4K   r1=r1-20:r2=r2-20
r1   IF r1=100 THEN r1=200
MY   IF r2=300 THEN r2=500
BT   FOR 1=50 TO 0 STEP-10
```

Listing. Ein spannendes Spiel mit toller Grafik. Bitte mit dem Checksummer eingeben.

```
K8   FOR a=r1 TO r2 STEP 75
g3   SOUND a,1.4,1,0
Y2   NEXT:NEXT
7e   GOTO KEY5
Vp0 gitter:
381  FOR a=5 TO 305 STEP 20
B7   LINE(a,5)-(a,180),16:NEXT
CE   FOR b=5 TO 180 STEP 20
VQ   LINE(5,b)-(305,b),16:NEXT
Lx   RETURN
660 rahmen:
DK1  PAINT(0,0),5:LINE(5,5)-(305,180),4,bf
7c   LINE(3,3)-(308,183),2,b
P1   RETURN
Sk0 rahmen2:
1X1  LINE(5,5)-(305,180),5,b
nN   PAINT (0,0),5
1T   LINE  (3,3)-(308,183),2,b
U6   RETURN
830 rechtm:
pn2   FOR a=1 TO 320 STEP 20
jZ    FOR b=1 TO 200 STEP 10
kW    LINE(a,b)-(a+10,b-10),farbe,b
gR    LINE(a+10,b-5)-(a+20,b-15),farbe,b
mC    NEXT:NEXT:RETURN
AA0 sechsm:
vt2   FOR a=1 TO 320 STEP 20
pf    FOR b=1 TO 200 STEP 10
SU    CIRCLE(a,b),5,go
Ex    LINE(a+5,b)-(a+5,b+3),go1
Ga    LINE(a+5,b+10)-(a+5,b+7),go2
tJ    NEXT:NEXT:RETURN
Q30 level1:
AL   GOSUB rahmen:GOSUB gitter
sK   y1=160:y2=160
sD   points:
8J1  FOR a=1 TO 80
AH   PSET(RND*320,RND*200),6
PQ   NEXT:RETURN
ZD0 level2:
841  hg=12:dg=13:we=2:go=1:GOSUB quadrato
TZ   GOSUB rahmen2
mL   a=160:b=100
6T0 Kreis:
La1  CIRCLE(a,b),50,15
B7   CIRCLE(a,b),49,6:PAINT(a,b),6
ul   CIRCLE(a,b),39,4:PAINT(a,b),4
zL   CIRCLE(a,b),39,15
xI   LINE(a-50,b-6)-(a+50,b+6),4,bf
5M   LINE(a+4,b-42)-(a-4,b+42),4,bf
Oc   RETURN
nS0 level3:
cw1  GOSUB gitter: GOSUB rahmen2
Oq0 Muenzen:
uS1  x1=140:x2=160
G3   FOR a=20 TO 290 STEP 30
QD   FOR b=50 TO 150 STEP 90
1d   CIRCLE (a,b),10,1
Gq   PAINT (a,b),1 :CIRCLE(a,b),10,15
L1   NEXT:NEXT:RETURN
ye0 level4:
In1  x1=30:x2=270
SC   y1=100:y2=100
fT   GOSUB rahmen
V4   GOSUB gitter
NR0 BLOECKE:
IT1  FOR a=9 TO 305 STEP 40
QY   FOR b=9 TO 180 STEP 40
x9   LINE(a,b)-(a+12,b+12),15,bf
2N   LINE(a+2,b+2)-(a+10,b+10),1,bf
XE   LINE(a+4,b+4)-(a+8,b+8),2,bf
Xx   NEXT:NEXT:RETURN
Ct0 level5:
YP1  y1=80:y2=80
aI   hg=18:dg=17:we=2:go=7:GOSUB quadrato
UM   GOSUB Muenzen
Q1   n=7:k=2:l=19:m=6:n=24
FO   GOSUB diamant:CIRCLE(150,70),30,4
Wm   PAINT(150,70),4
oB   CIRCLE(150,70),30,6
zY   LINE(130,20)-(132,70),4,bf
K1   LINE(170,20)-(172,70),4,bf
g6   GOSUB rahmen2:RETURN
4r0 level6:REM 16
qb1  x1=100:x2=220
py   farbe=12
qB   GOSUB rechtm
e50 3 Baelle:
Jj1  FOR a=100 TO 280 STEP 120
sc   FOR b=40 TO 170 STEP 50
PE   CIRCLE(a,b),35,5:PAINT(a,b),5
F1   CIRCLE(a,b),25,4:PAINT(a,b),4
Cn   CIRCLE(a,b),32,1:CIRCLE(a,b),34,13
1V   NEXT:NEXT
if   FOR a=100 TO 300 STEP 120
78   LINE(a-10,50)-(a+10,170),4,bf:NEXT
uK   GOSUB rahmen2:RETURN
f00 level7:
J71  b1=2:b2=4:x1=40:x2=260
kE   PAINT (13,13),4:FOR a=1 TO 100:PSET(RND*320,RND*200),10
Xc   NEXT
p90 GOSUB gitter :GOSUB rahmen2
vr   KNOCHEN:
ts1  CIRCLE(150,100),40,5:PAINT(150,100),5
Fo   LINE(60,90)-(240,110),5,bf
Ah   CIRCLE(40,100),30,5:PAINT(40,100),5
Jz   CIRCLE(40,100),27,4:PAINT(40,100),4
22   CIRCLE(250,100),30,5:PAINT(250,100),5
JS   CIRCLE(250,100),27,4:PAINT(250,100),4
7D   CIRCLE(150,100),35,4:PAINT(150,100),4
Uf   LINE(19,95)-(275,110),4,bf
zb   RETURN
fI0 level8:REM 8
8L1  x1=20:x2=260:y1=160:y2=160
eq   h=4:k=2:l=19:m=6:n=24
J2   PAINT(0,0),h:FOR o=1 TO 300
zp   PSET(RND*320,RND*200),2:NEXT
g3   go=4:go1=9:go2=9:GOSUB sechsm
jh   GOSUB rahmen2:GOSUB diamant
Ps   GOSUB weg:RETURN
Sr0 diamant:
SS1  FOR a=70 TO 110
HX   LINE(150,130)-(a,70),k
kP   LINE(150,130)-(a+40,70),l
7m   LINE(150,130)-(a+80,70),m
eQ   LINE(150,130)-(a+120,70),n
x2   NEXT
K3   FOR b=70 TO 230 STEP 40
Vo   LINE(150,130)-(b,70),m
05   NEXT
jJ   FOR w=70 TO 110
fA   LINE (150,20)-(w,69),k
sm   LINE (150,20)-(w+40,69),l
F9   LINE (150,20)-(w+80,69),m
uv   LINE (150,20)-(w+120,69),n
yz   NEXT:RETURN
rw0 weg:
yV1  LINE(70,20)-(230,40),4,bf
3c   LINE(117,40)-(185,40),6
Os   LINE(185,40)-(230,69),6
S4   RETURN
Q10 muster:
nl1  FOR a=1 TO 320 STEP 20
ja   FOR b=1 TO 200 STEP 20
FE   CIRCLE(a,b),10,go
7G   LINE(a-10,b-10)-(a+10,b+10),go
CL   LINE(a+10,b-10)-(a-10,b+10),go
1B   NEXT:NEXT:RETURN
pi0 muster2:
wv1  FOR a=1 TO 320 STEP 30
sk   FOR b=1 TO 200 STEP 30
ML   CIRCLE(a,b),10,go
EN   LINE(a-10,b-10)-(a+10,b+10),go
JS   LINE(a+10,b-10)-(a-10,b+10),go
VA   CIRCLE(a+10,b+10),10,go
be   LINE(a-20,b-20)-(a+20,b+20),go
gJ   LINE(a+20,b-20)-(a-20,b+20),go
vL   NEXT:NEXT:RETURN
6d0 level9:REM 1
Zd   DISKETTE:
c41  PAINT(0,0),17
x4   go=2:GOSUB muster:GOSUB rahmen2
TO   LINE(100,30)-(220,150),4,bf
B9   LINE(100,30)-(220,150),15,b
```

```
Cx   LINE(100,69)-(220,81),15,b
cQ   LINE(100,70)-(220,80),28,bf
YH   LINE(101,50)-(106,44),17,bf
He   LINE(110,40)-(160,60),2,bf
D3   CIRCLE(160,90),20,17
ZR   PAINT(160,90),17
tK   FOR r=20 TO 18 STEP-1
US   CIRCLE(160,90),r,15,2.6,.485,1
hm   NEXT
NO   CIRCLE(190,100),3,17
Kh   PAINT (190,100),17
1d   RETURN
wp0 level10:
Q91  x1=145:x2=175
uM   PAINT(0,0),17
hn   GOSUB rahmen2
oP   s=4:g=6:hor=30:hor2=280:ver=40:ver2=180
jN   schritt=30
7L0 gebilde:
bX1  FOR a=hor TO hor2 STEP schritt
6M   FOR b=ver TO ver2 STEP schritt*2
TA   LINE(a,b)-(a+20,b-20),g,bf
fH   CIRCLE(a+10,b-10),10,s
fU   LINE(a,b)-(a+20,b-20),s
EB   LINE(a+20,b)-(a,b-20),s
70   PAINT(a+10,b-11),s
Ss   NEXT:NEXT:RETURN
E80 level11:
Mh1  y1=170:y2=170:x1=50:x2=80
b7   PAINT(0,0),13:GOSUB rahmen2:GOSUB gitter
vP   FOR a=170 TO 290 STEP 20
1I   CIRCLE(a,90),5,6:PAINT(a,90),6
X7   NEXT:gr=15:ge=9:we=2:s=4
8f   hor=150:ver=90
0d0 PACMAN:
hp1  CIRCLE(hor,ver),50,gr,.875,5.495,1
ps   CIRCLE(hor,ver),49,ge,.875,5.495,1
NL   LINE(hor,ver)-(hor+29,ver-37),ge
Dj   LINE(hor+2,ver)-(hor+32,ver-38),gr
3m   LINE(hor,ver)-(hor+33,ver+34),ge
5p   LINE(hor+2,ver)-(hor+36,ver+35),gr
Mm   PAINT(hor-10,ver),ge
Xy   CIRCLE(hor,ver-30),5,4:PAINT(hor,ver-30),s
p1   CIRCLE(hor,ver-30),3,we
YA   RETURN
ZU0 level12:
Ux1  PAINT(0,0),18
Om   hor=60:ver=60
sm   gr=7:ge=9:we=2:s=4:GOSUB PACMAN
s6   hor=160:ver=120:GOSUB PACMAN
A2   hor=260:ver=60:GOSUB PACMAN
tE   GOSUB gitter:GOSUB rahmen2
6m   s=17:g=6:hor=26:hor2=280:ver=166:ver2=176
02   schritt=40:GOSUB gebilde
iK   RETURN
mi0 level13:
7j1  PAINT(0,0),8:
NT   GOSUB rahmen2
EH   g=6:r=5:gr=15:rad=20:hor=105:hor2=235
eL   schritt=60
TS0 ATOMS:
HD1  FOR a=hor TO hor2 STEP schritt
nm   CIRCLE(a,45),rad,r,,,.2
zn   CIRCLE(a,45),rad,r,,,5
NN   CIRCLE(a,45),rad,gr
mv   CIRCLE(a,155),rad,r,,,.2
nl   CIRCLE(a,155),rad,r,,,5
gN   CIRCLE(a,155),rad+4,gr
6S   PAINT(a,45),r
or   PAINT(a,155),r:NEXT
1P   FOR b=hor-40 TO hor2-80 STEP schritt
jC   CIRCLE(45,b),rad,g,,,.2
8q   CIRCLE(275,b),rad,g,,,.2
wE   CIRCLE(45,b),rad,g,,,5
Ah   CIRCLE(275,b),rad,g,,,5
cB   PAINT(45,b),g
m2   PAINT(275,b),g
gh   NEXT:RETURN
C90 level14:
LK1  x1=20:x2=40
sW   go=21:GOSUB muster:GOSUB rahmen2
ms   gr=15:bl=13:r=5
SG0 COMMODORE:
ik1  CIRCLE(150,90),81,gr,1.57,4.71,1
h1   CIRCLE(150,90),80,bl,1.57,4.71,1
ct   CIRCLE(150,90),50,bl,1.57,4.71,1
se   LINE(149,10)-(149,40),bl
1S   LINE(150,10)-(150,40),gr
Fz   LINE(149,140)-(149,170),bl
s8   LINE(150,140)-(150,170),gr
5B   CIRCLE(150,90),49,gr,1.57,4.71,1
jC   PAINT(80,90),bl
S6   LINE(150,50)-(230,80),bl,bf
ta   LINE(150,50)-(230,80),gr,b
9c   LINE(150,100)-(230,130),r,bf
Ur   LINE(150,100)-(230,130),gr,b
00   RETURN
YW0 level15:
KC1  farbe=8:x1=30:x2=270
h2   GOSUB rechtm
MU   FOR a=5 TO 315 STEP 40
t3   FOR b=5 TO 195 STEP 40
aN   PAINT(a,b),8
pJ   NEXT:NEXT
CO   GOSUB BLOECKE:GOSUB rahmen2:RETURN
j10 level16:
4H1  go=7:x1=140:x2=165
ib   GOSUB muster2:GOSUB rahmen2
Da   a=54:b=100:GOSUB Kreis
BQ   a=154:b=100:GOSUB Kreis
GW   a=254:b=100:GOSUB Kreis
dF   RETURN
zc0 quadrato:
yw1  FOR a=1 TO 320 STEP 20
ul   FOR b=1 TO 200 STEP 20
W1   LINE(a,b)-(a+15,b+15),hg,bf
iR   LINE(a+2,b+2)-(a+13,b+13),dg,bf
gy   LINE(a,b)-(a+15,b+15),we
ob   LINE(a,b+15)-(a+15,b),we
HB   LINE(a+5,b+5)-(a+10,b+10),go,bf
y0   NEXT:NEXT:RETURN
3q0 level17:REM 17
eC1  x1=140:x2=160
81   hg=27:dg=28:we=28:go=1
gt   GOSUB quadrato
TZ   GOSUB rahmen2
nM   b1=4:b2=2
Cz   GOSUB KNOCHEN
uW   RETURN
DE0 level18:
Ez2  x1=100:x2=220
Yz   s=12:g=13:hor=30:hor2=280
2T   ver=40:ver2=180:schritt=20
fv   GOSUB gebilde:GOSUB 3Baelle
0c   RETURN
M00 level19:
dG2  hg=15:dg=27:we=2:go=1
N0   GOSUB quadrato:GOSUB rahmen2
B0   x1=139:x2=159:RETURN
1v0 level20:
Xi2  GOSUB rahmen:GOSUB gitter
1o   LINE(40,40)-(100,160),6,bf
KD   LINE(50,50)-(90,150),4,bf
zp   LINE(43,43)-(97,157),1,b
G1   LINE(200,40)-(260,160),5,bf
Dh   LINE(210,50)-(250,150),4,bf
Jz   LINE(203,43)-(257,157),1,b
W1   x1=70:x2=230
Eq   RETURN
E90 level21:
X62  GOSUB gitter
rm   COLOR 6
GW   LOCATE 3,7:PRINT"TROFF PROGRAMMED BY"
JX   LOCATE 7,13:PRINT"THANKS TO"
IG   LOCATE 11,10:PRINT"PROGRAMMED FOR"
sm   COLOR 5
Gb   LOCATE 5,6:PRINT"PATRICK KÄLIN IN 1987"
4J   LOCATE 9,10:PRINT"KUSI AND HILLE"
1g   LOCATE 13,10:PRINT"AMIGA MAGAZIN"
Zz   GOSUB rahmen2:RETURN
(C) 1988 M&T
```

Listing. Ein spannendes Spiel mit toller Grafik. Bitte mit dem Checksummer eingeben. (Schluß)

Spionage auf der Diskette

Für einen Diskettenmonitor gibt es viele Anwendungen, zum Beispiel das Suchen von Paßwörtern bei Spielen oder das Verändern von einzelnen Bytes in Dateien. »DiskSpy« hilft Ihnen beim Spionieren.

Grundvoraussetzung für die sinnvolle Anwendung von Diskspy sind Grundkenntnisse über den Diskettenaufbau. Außerdem sollten Sie aus Vorsichtsgründen Manipulationen nur auf einer Kopie Ihrer Diskette vornehmen. Wie schnell ist eine Datei zerstört, die Sie noch unbedingt benötigen.

Das Programm läßt sich von der Workbench durch Doppelklick auf das Icon oder vom CLI aus aufrufen. Im letzteren Fall starten Sie DiskSpy mit dem Befehl

```
DiskSpy
```

Bei beiden Alternativen müssen Sie allerdings vorher den Speicher über der 512-KByte-Grenze ausschalten. Dazu verwenden Sie das Programm »NoFastMem« von Ihrer Workbench-Diskette. Sollten Sie dies vergessen, stürzt das Programm zwar nicht unbedingt ab, aber es gibt seltsame Effekte.

Doch nun zur Bedienung des Programms. DiskSpy wird über die Tastatur bedient, was bei solchen Werkzeugen auch sinnvoll ist. Achten Sie bei der Befehlseingabe bitte auf die notwendigen Leerstellen zwischen den Parametern.

Der wichtigste Befehl zuerst:

l — List Helptext

Mit diesem Kommando erreichen Sie, daß eine Liste sämtlicher Befehle auf dem Bildschirm ausgegeben wird. Dies ist am Anfang sehr hilfreich und erleichert die Arbeit ungemein.

x — Exit

Zum Verlassen des Programms müssen Sie dann noch die Sicherheitsabfrage »Are you sure (y/n) ?« mit <y> beantworten. Mit <n> bleiben Sie in DiskSpy. Drücken Sie eine andere Taste, erscheint erneut die Abfrage.

d — Do sum

Bevor ein veränderter Sektor wieder auf die Diskette geschrieben werden darf, muß man die Checksumme berechnen lassen. Dies dürfen Sie nie vergessen, da sonst der Sektor nicht mehr lesbar ist. Nur mit viel Aufwand läßt sich so eine Diskette wieder komplett herstellen.

h — Change Head

Der Amiga beschreibt seine Disketten beidseitig. DiskSpy erlaubt es nun mit diesem Befehl zwischen den zwei Schreib-/Leseköpfen hin- und herzuschalten. Wenn also bei der Suche nach Daten mit dem »f«-Befehl etwas nicht gefunden wird, probieren Sie die andere Diskettenseite aus. Durch den Aufruf gelangen Sie immer auf die andere Seite. Nach zweimaligem Eingeben sind Sie also wieder auf der ersten Seite der Diskette.

+ — Next Cylinder

Mit diesem Aufruf positionieren Sie die Schreib-/Leseköpfe auf die nächste Spur. Stehen diese schon auf dem letzten Track (79), so wird die Spur, der Sektor, der Kopf und die Blocknummer ausgegeben.

- — Back Cylinder

Dies ist das Gegenstück zum Kommando »+«. Hiermit gehen Sie auf den vorherigen Zylinder. Steht der Kopf schon auf Spur 0, so werden analog zu oben die Werte auf den Bildschirm gebracht.

s — Show Cylinder,Sector,Head and Block

Wenn Sie nun nach einigem Hin und Her auf der Diskette nicht mehr wissen, wo dem Laufwerk der »Kopf steht«, hilft Ihnen dieser Befehl. Er gibt die Spur, den Sektor, die Diskettenseite sowie die Blocknummer aus.

c — Cli-Command,c again-get back to DiskSpy

Damit Sie nicht ein eigenes CLI-Fenster benötigen, können Sie von DiskSpy aus auch CLI-Befehle aufrufen. Nach Eingabe von <c> erscheint das Prompt 1>. Wie im CLI gewohnt, können Sie nun Kommandos eingeben. Nach Beendigung gelangen Sie durch erneutes Eintippen von <c> wieder in den normalen Modus zurück. Auch das Prompt ist wieder der von DiskSpy gewohnte Punkt.

r tt ss — Read Cylinder(0-79) Sektor(0-10)

Sie wollen aber sicher nicht nur den Kopf über die Diskette bewegen, sondern auch Änderungen vornehmen. Der erste Schritt ist dabei das Lesen eines Sektors. Bei dem Aufruf geben Sie die Spur- und Sektornummer an, von wo Sie lesen wollen. Das Kommando

```
r 40 00
```

liest den Inhalt des ersten Sektors vom vierzigsten Track in den Puffer.

w tt ss — Write Cylinder(0-79) Sektor(0-10)

Dies ist das Gegenstück zum Befehl »r«. Mit ihm schreiben Sie die Daten im Puffer in den angegebenen Sektor.

t tt — Show Cylinder(0-79)

Alle Sektoren der angegebenen Spur erscheinen nacheinander auf dem Bildschirm. Wie im CLI üblich, können Sie die Ausgabe anhalten und unterbrechen.

f tt ll "text" — Find text between Cylinder tt and Cylinder ll

Ein weiterer interessanter Befehl, der sicher oft seine Anwendung findet. Die Suche nach Texten auf der Diskette ist damit leicht möglich. Mit dem Befehl

```
f 00 79 "library"
```

suchen Sie von Spur 0 bis 79 nach dem Text »library«. Beachten Sie, daß der Text nur gefunden wird, wenn er auf der aktuell eingestellten Seite steht. Falls DiskSpy den Suchtext nicht findet, schalten Sie auf den anderen Kopf um und suchen alle Spuren durch.

o mmmm nnnn dd — Fill buffer from mmmm to nnnn with dd

Der vorhin schon angesprochene Puffer enthält einen kompletten Sektor. Wenn Sie diesen Speicherbereich oder Teile davon mit einem Wert füllen wollen, hilft Ihnen dieses Kommando. Der Puffer hat eine Größe von 512 Byte (0000 bis 01ff). Die Angaben müssen sich daher auch auf diesen Bereich beziehen. Mit

```
o 0050 0100 41
```

füllen Sie zum Beispiel den Bereich von 0050 (dezimal 80) bis 0100 (dezimal 160) mit dem Wert 41 (dezimal 65, das Zeichen A).

m mmmm nnnn or m — Memorydump from mmmm to nnnn or from 0000 to 01ff

Um sich den Puffer anzuschauen, benutzen Sie diesen Befehl. Geben Sie beide Grenzen an, wird genau dieser Bereich ausgegeben. Übergeben Sie nur eine Adresse, erscheint nur eine Ausgabezeile mit Daten. Verwenden Sie gar keinen Parameter, sehen Sie den ganzen Puffer.

e mmmm — Edit from mmmm

Nachdem Sie nun die zu editierenden Daten gefunden haben, können Sie diese verändern. Dazu geben Sie eine Adresse an, an der Sie beginnen wollen. Es erscheint die komplette Zeile und darunter noch einmal die gewünschte Adresse. Hier können Sie nun die Werte eintragen. Dabei müssen Sie darauf achten, daß Sie die Leerstellen zwischen den einzelnen Angaben nicht vergessen. Haben Sie alle gewünschten Bytes editiert, drücken Sie <RETURN>. Es ist also nicht notwendig, die gesamte Zeile einzugeben.

VESALIA Top Angebote

AMIGA-ZUBEHÖR

3,5" Slimline Laufwerk NEC 1036a
abschaltbar, mit durchgeführtem Bus, Metallgeh. u. Frontblende Amigafarben **339,–**

3,5" Slimline Laufwerk CHINON
abschaltbar, mit durchgeführtem Bus, Metallgeh. Amigafarben **299,–**

3.5" internes Laufwerk
für Amiga 2000 bereits einbaufertig modifiziert, mit Einbauanleitung **239,–**

3,5" Super-Slimline Laufwerk
abschaltbar m. durchgef. Bus, Metallgeh. Amigafarben **295,–**

5,25" Laufwerk CHINON
40/80 Track schaltbar und abschaltbar, Amiga- und MS-DOS kompatibel, Metallgeh. u. Frontblende Amigafarben **ab 369,–**

3,5" Doppellaufwerk 2× NEC 1036a
Laufwerke einzeln abschaltbar **595,–**

Gehäuse f. NEC 1035 bzw. Teak FD 35 FN
m.Ausspar.f.Ein-Ausschalter, durchg.Port **19,50**

64er Emulator (Ready Soft) **149,–**

Ersatzkabel A500/A2000 **26,–**

Ersatzkabel A1000 **26,–**

Monitorverbindung **26,–**

PAL-Genlock stellt die Verbindung zw. AMIGA und Videorekorder dar! **nur DM 548,–**

512 KB Karte für A 500, mit akkugep. Uhr, 512 KB, abschaltb., alle Baust. gesockelt **269,–**

2 MB Rambox
für Amiga 500, abschaltbar, mit durchgef. Bus, Metallgehäuse Amigafarben **948,–**

2 MB Golem Rambox
für Amiga 1000, abschaltbar, mit durchgef. Bus, Metallgehäuse Amigafarben **948,–**

2 MB intern
für A 2000 (orig. Commodore) einbaufertig **850,–**

20 MB Festplatte für Amiga 2000 **1395,–**

Amiga 1000-Uhr
steckbar an Druckerport (durchgeführt) **69,–**

EASYL Grafik und Zeichentablett für A500 **678,–**

EASYL Grafik und Zeichentabl. für A1000 **798,–**

EASYL Grafik und Zeichentabl. für Amiga 2000 arbeitet z. B. mit Page-Setter, Deluxe Paint II AEGIS Images **798,–**

DSOUND V 2.2 Plus für Amiga 1000 kompl. im Gehäuse mit Anleitung und Software **198,–**

DSOUND V 2.2 Plus für Amiga 500 und 2000 kompl. im Gehäuse mit Anleitung und Software **228,–**

Digi View PAL mit dtsch. Handbuch **279,–**

Mousepath für die schnelle Maus **16,90**

VESALIA AMIGA-Software

FAST LIGHTNING
Der neue Maßstab, das zur Zeit schnellste Kopierprogramm. In nur 66 Sekunden können drei Sicherheitskopien gleichzeitig erstellt werden. Fast Lightning kopiert in 4 Modi bis Track 81.
89,–

TURBOCOPY
Das bewährte Kopierprogramm zum Erstellen von Sicherheitskopien mit zwei Laufwerken.
59,–

BOOTBLOCK-GENERATOR
Zum Erstellen eines eigenen Vorspanns mit Sound, sehr einfach zu bedienen. **49,–**

WHITE LIGHTNING
Das schnellste Kopierprogramm für nur ein Laufwerk **49,–**

Bitte erfragen Sie für Speichererweiterungen die aktuellen Preise und Lieferzeiten.

Händleranfragen erwünscht

Vesalia Versand
Soft- und Hardware
Entwicklung · Produktion
G. Does, 4230 Wesel, Tel. 0281/65466
Nur Versand: Marienweg 40

Vesalia-Produkte erhalten Sie auch in ...

Niederlande:
E. C. R. bc, Postbus 635,
7500 AP. Enschede
Tel. 053/762884

Österreich und Schweiz:
INTERCOMP A. Mayer,
Heldendankstr. 24,
6900 Bregenz,
Tel 05574/27344

Belgien:
Elcatron
Fruithoflaan 105, Bus 42
B-2600 Berchem
Tel. 03/4498926

?# or ?$ — Compute to dez(0-65535) or to hex($0-ffff)
Da im gesamten Programm nur mit Hexadezimalzahlen gearbeitet wird, stellt DiskSpy noch eine Hilfe zur Verfügung. Diese Befehle berechnen das dezimale beziehungsweise hexadezimale Äquivalent der eingegebenen Zahl. Mögliche Aufrufe sind zum Beispiel:

```
?#00123
?$0f0f
```

Es müssen fünf Stellen bei den dezimalen und vier Stellen bei den hexadezimalen Zahlen übergeben werden.

Sie besitzen nun also ein Werkzeug, mit dem Sie Daten direkt auf Diskette bearbeiten können. Seien Sie dabei aber vorsichtig, denn sonst sind schnell wichtige Daten verloren.

Einige sinnvolle Anwendungen finden Sie auch in den Tips & Tricks »Patchen mit System« auf den Seiten 108 und 109.

(Michael Stein/rb)

Programmname:	DiskSpy
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Assembler
Assembler:	Seka-Assembler

```
Programm : DiskSpy
------------------

2TO ;******* DiskSpy V1.0 ********
70  ;* () 1987 by Michael Stein *
sq  ;* 3.9.87-24.10.87          *
AP  ;* für AMIGA                *
zG  ;****************************
Aa  execbase = 4
ZR  ;** exec-routinen **
x2  findtask    =-294
8M  addport     =-354
Am  remport     =-360
YI  oldopenLib  =-408
72  closelib    =-414
bU  opendevice  =-444
vy  closedev    =-450
pE  sendio      =-462
xr  doio        =-456
tp  waitport    =-384
FO  getmsg      =-372
Oi  ;** dos-routinen **
R3  Open    =-30
t8  Close   =-36
8x  read    =-42
gJ  Write   =-48
nt  ioerr   =-132
m0  exit    =-144
FS  execute =-222
x1  start:
ms1  move.l execbase,a6
3k   sub.l a1,a1
BT   jsr findtask(a6)         ;eigenen T
     ask suchen
lo   move.l d0,readreply+$10 ;zum Task
     sperren in diskio
Ik   lea readreply,a1
Zk   jsr addport(a6)          ;Port such
     en
kj   move.l readreply+$10,a4
Py   tst.l $ac(a4)
4r   bne auscli               ;Aus dem C
     LI gestartet ?
eD   lea $5c(a4),a0           ;Nein,dann
     Workbench
1M   jsr waitport(a6)
y3   lea $5c(a4),a0
y4   move.l execbase,a6
WB   jsr getmsg(a6)       ;Message ab
     holen
c90 auscli:
171  move.l execbase,a6
wW   lea dosname,a1
fy   clr.l d0
yB   jsr oldOpenLib(a6)  ;dos-library ö
     ffnen
pm   move.l d0,dosbase
Tk   beq error
4TO oeffnen:
NT1  move.l #fenstername,d1
h3   move.l #1005,d2
Nv   move.l dosbase,a6
Av   jsr open(a6)         ;Fenster öffne
     n
pY   tst.l d0
Ho   beq notdone
XE   move.l d0,handle
SO   move.l dosbase,a6
13   move.l handle,d1
I7   move.l #titel,d2
cI   move.l #titelende-titel,d3
r9   jsr write(a6)        ;Titel ausgebe
     n
Ug   bsr listhelp1
lt   bsr retout
d7   bsr show1
ua0 trdopen:
OU1  move.l execbase,a6
1L   lea diskio,a1
OE   clr.l d0             ;df0
7R   clr.l d1
Hx   lea trddevice,a0
A2   jsr opendevice(a6)  ;trackdiskdevi
     ce öffnen
7q   tst.l d0
Ei   bne notdone
Eu0 loop1:                     ;Eingabe
     Tastatur
fQ1  bsr clearline
Aa   move.l dosbase,a6          ;**DiskS
     py prompt ausgeben**
KM   move.l handle,d1
Sv   move.l #point,d2
Yq   move.l #pointend-point,d3
5K   jsr write(a6)
pw   moveq #80,d3
Tv   bsr readchr
J5   cmp.b #$0a,inline   ;nur Return ?
zX   beq.s loop1         ;dann zurück
rV   lea inline,a0
Fe   clr.b 0(a0,d0)      ;Null-byte ans
      ende
k2   lea locate,a0
bE   clr.b 0(a0,d0)      ;Auch bei Loca
     te
8A   cmp.b #"x",inline   ;Buchstaben fü
     r Befehle
Fl   beq ende            ;Suchen und ve
     rzweigen
7U   cmp.b #"r",inline
lc   beq readtrack
YO   cmp.b #"w",inline
5A   beq writetrack
Sg   cmp.b #"+",inline
tj   beq plustrack
eu   cmp.b #"-",inline
KE   beq minustrack
q8   cmp.b #"m",inline
21   beq memory
Mk   cmp.b #"s",inline
86   beq show
Ts   cmp.b #"t",inline
za   beq tracker
NY   cmp.b #"f",inline
hC   beq find
tA   cmp.b #"l",inline
04   beq listhelp
AU   cmp.b #"o",inline
4X   beq occupy
EM   cmp.b #"c",inline
Vn   beq.s CLI
Qa   cmp.b #"e",inline
M1   beq edit
NW   cmp.b #"d",inline
Nv   beq summenbildung
Qy   cmp.b #"?",inline
j6   beq wandler
ly   cmp.b #"h",inline
Rm   beq headset
U2   move.l dosbase,a6
35   move.l handle,d1
oK   move.l #was,d2
zS   move.l #wasend-was,d3
md   jsr write(a6)         ;Unbekanntes
      Kommando
91   bra loop1   ;Zurück zur Eingabe
c50 CLI:                        ;Cli C
    ommando
b91  move.l dosbase,a6
AC   move.l handle,d1
LV   move.l #prompt,d2
Km   move.l #promptend-prompt,d3
q9   jsr write(a6) ;* 1> ausgeben
7d   moveq #80,d3  ;80 zeichen
Jl   bsr readchr
fJ   lea inline,a0
M3   clr.b 0(a0,d0)
em   cmp.b #"c",inline
Bs   beq loop1
CW   cmp.b #$0a,inline  ;nur return
xF   beq.s CLI
e4   cmp.b #" ",inline  ;oder space
v0   beq.s CLI          ;ja,dann zurück
q0   move.l dosbase,a6
wq   move.l #inline,d1
Pk   clr.l d2
VZ   move.l handle,d3
dc   jsr execute (a6)
gv   bra.s CLI
URO ende:                 ;x=exit
BQ1  bsr are_you_sure
lr   move.l execbase,a6
Oi   lea diskio,a1
Co   move #9,28(a1)  ;Motor
kD   clr.l 36(a1)    ;aus
hQ   jsr sendio(a6)
qw   move.l execbase,a6
Jl   lea readreply,a1
ZQ   jsr remport(a6)       ;Replyport lö
     schen
tz   move.l execbase,a6
Wq   lea diskio,a1
5T   jsr closedev(a6)      ;Trddevice sc
     hließen
9h   move.l dosbase,a6
ik   move.l handle,d1
zM   jsr close(a6)         ;Fenster schl
     ießen
510 notdone:
yR1  move.l dosbase,a1
17   move.l execbase,a6
gH   jsr closelib(a6)      ;dos-library
     schließen
MWO error:
Hp1  move.l dosbase,a6
15   clr.l d1
1D   jmp exit(a6)   ;Return ohne Fehler
rs   rts
IVO retout:                    ;** gibt Ret
    urn aus **
Mu1  move.l dosbase,a6
vx   move.l handle,d1
ht   move.l #ret,d2
y7   move.l #retend-ret,d3
gv   jsr write(a6)
yz   rts
UJO clearline:        ;** Eingabepuffer
     loeschen **
L21  move.l #79,d0
```

```
183 Ia   lea locate,a0
184 WB   lea inline,a1
185 eQ0 c1:
186 kN1  clr.b (a0)+
187 qU   clr.b (a1)+
188 ko   dbra d0,c1
189 78   rts
190 Uk0 readchr:            ;** liest Zeiche
        n von Tastatur ein **
191 ZY1  move.l #inline,d2 ;** Zeichen ab(d
         2) ablegen
192 dB   move.l dosbase,a6
193 CE   move.l handle,d1
194 br   jsr read(a6)
195 DE   rts
196 x40 plusstrack:       ;nächster track,b
        ei gleichem sektor
197 301  cmp.l #79,track
198 ge   beq show
199 yI   add.l #1,track
200 cz   bsr readtrack1
201 nR   bra loop1
202 g70 minustrack:    ;letzter track
203 Fv1  cmp.l #0,track
204 mk   beq show
205 LC   sub.l #1,track
206 FB   bsr.s readtrack1
207 tX   bra loop1
208 2M0 readtrack:       ;Track lesen
209 Ah1  bsr tracksearch
210 41   bsr sektorsearch
211 KG   bsr.s readtrack1
212 yc   bra loop1
213 zq0 readtrack1:    ;Track ohne anderer
        ext. lesen
214 Q81  move.w #2,modus
215 jE   bsr atrack
216 YZ   rts
217 ZY0 writetrack:    ;Track schreiben
218 Jq1  bsr tracksearch
219 Du   bsr sektorsearch
220 oS   cmp.b #11,sektor
221 Wx   bcc loop1
222 Kp0 write1:
223 Mb1  bsr are_you_sure
224 f0   move.w #3,modus
225 Uu   bsr.s atrack
226 mW   move.w #4,modus
227 Ww   bsr.s atrack
228 Es   bra loop1
229 DC0 atrack:   ;Diskio ausführen
230 Na1  bsr diskin
231 fz   lea diskio,a1
232 Vc   move.l #readreply,14(a1)
233 ng   move.w modus,28(a1)
234 dj   move.l #diskbuff,40(a1)
235 zi   move.l #512,36(a1)
236 8E   move.l execbase,a6
237 T1   bsr.s blockrechnen
238 I6   move.l d1,44(a1)
239 01   jsr doio(a6)
240 wx   rts
241 FB0 blockrechnen:    ;Offset berechnen
242 9S1  move.l track,d1
243 VQ   mulu #22,d1
244 Gr   add.l sektor,d1
245 EN   move.l #11,d2
246 OV   move.l head,d3
247 SC   mulu d3,d2
248 oq   add.l d2,d1
249 Dy   mulu #512,d1
250 kU   move.l d1,block
251 78   rts
252 2L0 diskin: ;** Disk eingelegt ?? **
253 1L1  lea diskio,a1
254 2z   move #14,28(a1)
255 RX   move.l execbase,a6
256 K3   jsr sendio(a6)
257 5P   lea diskio,a1
258 dh   move.l 32(a1),d7
259 Bb   cmp.l #0,d7
260 Qi   beq.s diskinserted
261 kI   move.l dosbase,a6
262 JL   move.l handle,d1
263 Z5   move.l #text6,d2
264 18   move.l #text6end-text6,d3
265 4J   jsr write(a6)
266 2A   bsr retout
267 rV   bra loop1
268 Rn0 diskinserted:
269 PQ1  rts
270 eE0 not_done:
271 uS1  move.l dosbase,a6
272 TV   move.l handle,d1
273 nK   move.l #text7,d2
274 2R   move.l #text7end-text7,d3
275 ET   jsr write(a6)
276 CK   bsr retout
277 1f   bra loop1
278 ux0 tracksearch:
279 mX1  move.b inline+3,d0
280 1b   move.b inline+2,d1
281 ZI   bsr wandel
282 H9   cmp.b #80,d1
283 Wx   bcc loop1
284 GA   move.l d1,track
285 fg   rts
286 TT0 wandel:
287 a81  sub #48,d0
288 hB   sub #48,d1
289 4w   mulu #10,d1
290 tZ   add d0,d1
291 1m   rts
292 b10 sektorsearch:
293 Gu1  move.b inline+5,d1
294 Js   move.b inline+6,d0
295 Xn   bsr.s wandel
296 8u   cmp.b #11,d1
297 kB   bcc loop1
298 s3   move.l d1,sektor
299 tu   rts
300 F40 tracker:
301 Fu1  bsr.s tracksearch
302 eR   clr.l sektor
303 ou0 tracker1:
304 9Y1  cmp.b #$39,$bfec01  ;Ctrl-Taste ?
305 M3   beq.s tracker2
306 Kh   bsr readtrack1
307 Y2   bsr show1
308 9W   bsr.s memo1
309 87   cmp.l #10,sektor
310 B4   beq tracker2
311 uW   add.l #01,sektor
312 zc   bra.s tracker1
313 290 tracker2:
314 cG1  bra loop1
315 aK0 memory:  ;Memorydump
316 jS1  cmp.b #$0a,inline+1
317 WV   beq.s me1
318 Nm   bsr.s memo3
319 hL   bra loop1
320 040 memo3:
321 Tu1  bsr two_4_digit
322 560 mem:
323 ec1  cmp.l #diskbuff+$0200,from
324 V2   bge loop1
325 DD   cmp.l #diskbuff+$0200,to
326 X4   bge loop1
327 uM   move.l from,a1
328 jQ   suba.l #diskbuff,a1
329 y9   move.w a1,startadr
330 3D   move.l from,memstart
331 x2   move.l to,memende
332 Zx   bsr.s memo2
333 RS   rts
334 PS0 me1:
335 ax1  bsr.s memo1
336 yc   bra loop1
337 5H0 memo1:
338 f81  move.l #diskbuff+$0200,memende
339 8j   move.l #diskbuff,memstart
340 d9   clr.w startadr
341 FN0 memo2:
342 EA1  move.l memstart,a4
343 WS0 m1:
344 Qd1  move startadr,d1
345 nI   add #16,startadr
346 1C   lea puffer,a0
347 0e   moveq #3,d3
348 8E   moveq #15,d4
349 fc0 m2:
350 6z1  rol.w #4,d1
351 6c   move d1,d2
352 H3   bsr nibble
353 pU   move.b d2,(a0)+
354 Tg   dbra d3,m2
355 Go   move.l dosbase,a6
356 pr   move.l handle,d1
357 nI   move.l #puffer,d2
358 Zh   move.l #pufferend-puffer,d3
359 ap   jsr write(a6)
360 NU   moveq #15,d5
361 rj   lea char,a1
362 eE   move.l a1,rette
363 wu0 m3:
364 UE1  lea byte,a0    ;in . oder Ascii um
         wandeln
365 om   move.l rette,a1
366 8u   move.b (a4),d1
367 oy   move.b d1,d2
368 VK   cmp.b #32,d1
369 sC   bcc.s inorder1
370 Z7   bra.s inorder2
371 nZ0 inorder1:
372 AV1  cmp.b #126,d1
373 ZK   bls.s inorder3
374 uh0 inorder2:
375 ho1  moveq #46,d1
376 0o0 inorder3:
377 Cr1  move.b d1,(a1)+
378 uU   move.l a1,rette
379 bP   move.b (a4)+,d1 ;Byte wandeln
380 Fq   lsr #4,d2
381 kW   bsr nibble
382 Ix   move.b d2,(a0)+
383 c8   move d1,d2
384 nZ   bsr nibble
385 L0   move.b d2,(a0)+
386 1J   move.l dosbase,a6
387 KM   move.l handle,d1
388 fy   move.l #byte,d2
389 i5   move.l #byteend-byte,d3
390 5K   jsr write(a6)
391 GR   dbra d5,m3
392 rP   move.l dosbase,a6
393 QS   move.l handle,d1
394 DX   move.l #char,d2
395 wv   move.l #charend-char,d3
396 BQ   jsr write(a6)
397 9H   bsr retout
398 zG   cmp.b #$39,$bfec01  ;Ctrl ?
399 ma   beq.s nichtmehr
400 fw0 mloop:
401 jJ1  cmp.b #$3b,$bfec01  ;Caps lock ?
402 Zr   beq.s mloop
403 G8   cmp.l memende,a4
404 Bp   bge.s nichtmehr
405 td   bne m1
406 Ss0 nichtmehr:
407 de1  rts
408 3F0 nibble:
409 hS1  and #$0f,d2
410 tf   add #$30,d2
411 LH   cmp #$3a,d2
412 C2   bcs.s ok
413 tt   add #7,d2
414 XR0 ok:
415 1m1  rts
416 xS0 listhelp:
417 Co1  bsr.s listhelp1
418 Iw   bra loop1
419 nT0 listhelp1:
420 Jr1  move.l dosbase,a6
421 su   move.l handle,d1
422 f0   move.l #helptext,d2
423 aL   move.l #helptextend-helptext,d3
424 ds   jsr write(a6)
```

Listing.
»DiskSpy«, ein Diskettenmonitor, geschrieben mit dem Seka-Assembler. Bitte mit dem Checksummer eingeben.

```
vw   rts
iDO  occupy:
Bc1  bsr two_4_digit
mC   bsr one_2_digit
MK   cmp.l #diskbuff+$0200,from
Dk   bge loop1
vv   cmp.l #diskbuff+$0200,to
Fm   bge loop1
Xy   move.l from,a0
6c   move.l to,a1
gW   cmpa.l a0,a1
1i   bcs loop1
ii   suba.l a0,a1
2I   move.l a1,d1
Wj   move.l with,d0
970  o1:
5i1  move.b d0,(a0)+
r8   dbra d1,o1
hL   bra loop1
JSO  two_4_digit: ;wandelt 2 4stellige H extexte in Hex
s51  moveq #3,d0
Pm   lea fbyte,a0
1m   lea inline+2,a1
LnO  tw1:
k21  move.b (a1)+,(a0)+
f1   dbra d0,tw1
yB   moveq #3,d0
1I   lea sbyte,a0
MC   lea inline+7,a1
VyO  tw2:
q81  move.b (a1)+,(a0)+
nu   dbra d0,tw2
ax   lea fbyte,a0
O3   bsr hwandel
ly   adda.l #diskbuff,d1
60   move.l d1,from
uR   lea sbyte,a0
S7   bsr hwandel
p2   adda.l #diskbuff,d1
Aj   move.l d1,to
Za   rts
TUO  one_2_digit:  ;wandelt 2stelligen h extext in hex
R91  move.b inline+12,dbyte
Zk   move.b inline+13,dbyte+1
a5   lea dbyte,a0
aF   bsr hwandel
JL   move.l d1,with
gh   rts
o30  hwandel:
ey1  clr.l d1
ULO  h1:
z31  bsr.s h3
5A   cmp #$10,d0
Dl   bcc.s h2
7g   lsl.l #4,d1
Mn   or.b d0,d1
nX   bra.s h1
eWO  h2:
rs1  rts
jcO  h3:
141  clr.l d0
ga   move.b (a0)+,d0
oS   sub.l #"a",d0
V5   bcc.s h4
ab   add #39,d0
smO  h4:
7x1  add #10,d0
O1   rts
kFO  show:
Zb1  bsr.s show1
XB   bra loop1
1GO  show1:
zZ1  move.b #32,dezpuff+4
Z7   move.l dosbase,a6
8A   move.l handle,d1
4V   move.l #text1,d2
1E   move.l #text1end-text1,d3
t8   jsr write(a6)
Mf   move.l track,d1
sQ   bsr.s deziout
gE   move.l dosbase,a6
FH   move.l handle,d1
Fh   move.l #text2,d2
FU   move.l #text2end-text2,d3
OF   jsr write(a6)
YX   move.l sektor,d1
zX   bsr.s deziout
nL   move.l dosbase,a6
MO   move.l handle,d1
1z   move.l #text11,d2
yY   move.l #text11end-text11,d3
7M   jsr write(a6)
bg   move.l head,d1
6e   bsr.s deziout
uS   move.l dosbase,a6
TV   move.l handle,d1
fA   move.l #text5,d2
o9   move.l #text5end-text5,d3
ET   jsr write(a6)
Sq   bsr blockrechnen
R1   divu #512,d1
bQ   bsr.s deziout1
XL   bsr.s deziout0
GO   bsr retout
bc   rts
unO  deziout:
pA1  lea dezpuff,a0
1S   divu #10,d1
s9   bsr.s ziffer
tA   bsr.s ziffer
Ki   clr.w dezpuff+2
SbO  deziout0:
Ck1  move.l dosbase,a6
1n   move.l handle,d1
n8   move.l #dezpuff,d2
7b   move.l #dezpuffend-dezpuff,d3
Wl   jsr write(a6)
op   rts
dnO  deziout1:
2N1  lea dezpuff,a0
Ut   divu #1000,d1
5M   bsr.s ziffer
Nq   divu #100,d1
70   bsr.s ziffer
2j   divu #10,d1
mx   bsr ziffer
4FO  ziffer:
9u1  add #$30,d1
yc   move.b d1,(a0)+
Ci   clr d1
Vx   swap d1
23   rts
onO  find:
nK1  bsr tracksearch
YC   move.b inline+5,d1
bA   move.b inline+6,d0
5o   bsr wandel
hR   cmp.b #80,sektor
2T   bcc loop1
AL   move.l d1,sektor
ug   move.l sektor,endtrack
uh   clr.l sektor
m2   cmpi.b #80,endtrack
7Y   bcc loop1
QJ   move.l track,d1
3I   cmp.l endtrack,d1
SE   bgt loop1
G4   lea locate,a1    ;Text gekennzeichn et durch ".."
kX   cmp.b #34,inline+8 ;ab Locate able gen
O6   beq.s fi1  ;(mit abschließendem NUl 1Byte
pT   bra loop1
HHO  fi1:
MI1  lea inline+9,a0
nWO  fi:
m91  move.b (a0)+,(a1)+
EI   cmp.b #34,(a0)
zb   bne.s fi
L3O  fiend:
wU1  move.l dosbase,a6
VX   move.l handle,d1
Z2   move.l #text3,d2
ct   move.l #text3end-text3,d3
GV   jsr write(a6)
1Z   move.l dosbase,a6
ac   move.l handle,d1
Ex   move.l #locate,d2
gG   move.l #locateend-locate,d3
La   jsr write(a6)
JR   bsr retout
LAO  f1:
zM1  bsr readtrack1
Dh   bsr show1
CT   cmp.b #$39,$bfec01  ;Ctrl ?
bI   beq loop1
thO  ;** ab hier findalgorithmus **
Dz1  lea diskbuff,a0
VLO  f2:
7Q1  lea locate,a1
aRO  f3:
bY1  move.b (a1),d0
ml   cmp.b (a0),d0
NC   beq.s f5
hZO  f4:
Eh1  adda.l #1,a0
UA   cmp.l #diskbuff+$0200,a0
yi   bne.s f3
Pj   bra.L f7
piO  f5:
YE1  cmp.l #diskbuff+$0200,a0
rE   beq.L f7
Lo   adda.l #1,a0
Pt   adda.l #1,a1
ol   move.b (a1),d0
zy   cmp.b (a0),d0
aP   beq.s f5
z1   cmp.b #0,(a1)
5o   bne.s f2
2wO  f6:
a81  move.l dosbase,a6
9B   move.l handle,d1
H1   move.l #text4,d2
Ng   move.l #text4end-text4,d3
u9   jsr write(a6)
sO   bsr retout
C7O  f7:
Fd1  add.l #1,sektor
KJ   cmp.l #10,sektor
ST   bls.s f8
zm   clr.l sektor
1r   move.l endtrack,d0
Xd   cmp.l track,d0
Du   beq loop1
2M   add.l #1,track
OKO  f8:
6w1  move.l endtrack,d0
ci   cmp.l track,d0
K1   bcs loop1
yA   bra f1
txO  edit:
g71  bsr two_4_digit
xO   move.l from,a0
JV   add.l #16,a0
z2   cmp.l #diskbuff+$01ff,a0
Jy   ble.s e1
p4   move.l #diskbuff+$01ff,a0
D1O  e1:
uP1  move.l a0,to
MT   bsr mem
24   sub #16,startadr
Qd   move startadr,d1
OB   lea puffer,a0
Nd   moveq #3,d3
7D   moveq #15,d4
ODO  e2:
5y1  rol.w #4,d1
5b   move d1,d2
G2   bsr nibble
oT   move.b d2,(a0)+
ot   dbra d3,e2
Fn   move.l dosbase,a6
oq   move.l handle,d1
mH   move.l #puffer,d2
```

Listing. »DiskSpy«, ein Diskettenmonitor, geschrieben mit dem Seka-Assembler (Fortsetzung)

```
Yg   move.l #pufferend-puffer,d3
Zo   jsr write(a6)
FO   bsr clearline
KR   moveq #80,d3
yQ   bsr readchr
v1   move.w startadr,a3
qB   add.l #diskbuff,a3
Q5   lea inline,a1
iY0 e3:
4a1  move.b (a1)+,dbyte
cD   move.b (a1)+,dbyte+1
FP   add.l #1,a1
yT   lea dbyte,a0
yd   bsr hwandel
tc   cmp.l #diskbuff+$0200,a3
Ho   bge loop1
Zq   cmp.b #$0a,inline
zg   beq loop1
Iz   move.b d1,(a3)+
54   cmp.b #$00,(a1)
At   bne.s e3
eI   bra loop1
5M0 summenbildung:
fR1  lea diskbuff,a0
tR   clr.l $14(a0)
x9   move.l #0,a1
AZ0 su1:
SS1  move.l (a0)+,d1
9D   suba.l d1,a1
tZ   cmp.l #diskbuff+$0200,a0
dv   bne.s su1
nZ   lea diskbuff,a0
MF   move.l a1,$14(a0)
qU   bra loop1
010 wandler:
gE1  cmp.b #"#",inline+1
KE   beq.s dezwandler
nM   cmp.b #"$",inline+1
OK   beq.s hexwandler
wa   bra loop1
6p0 hexwandler:
xV1  move.l dosbase,a6
WY   move.l handle,d1
yX   move.l #text9,d2
Jm   move.l #text9end-text9,d3
HW   jsr write(a6)
nE   bsr two_4_digit
uB   sub.l #diskbuff,from
Mr   move.l from,d1
o9   lea dezpuff,a0
LO   divu #10000,d1
Uf   bsr ziffer
Ih   divu #1000,d1
Wh   bsr ziffer
Be   divu #100,d1
Yj   bsr ziffer
qX   divu #10,d1
a1   bsr ziffer
bm   bsr ziffer
6a   bsr deziout0
Tb   bsr retout
Iw   bra loop1
S90 dezwandler:
Jr1  move.l dosbase,a6
su   move.l handle,d1
Go   move.l #text8,d2
Yz   move.l #text8end-text8,d3
ds   jsr write(a6)
n6   clr.l d0
sC   clr.l d1
xI   clr.l d2
20   clr.l d3
6v   move.b inline+2,d0
H3   move.b inline+3,d1
SB   move.b inline+4,d2
dJ   move.b inline+5,d3
oR   move.b inline+6,d4
49   sub.b #48,d0
8E   sub.b #48,d1
CJ   sub.b #48,d2
GO   sub.b #48,d3
KT   sub.b #48,d4
Op   mulu #10000,d0
1b   mulu #1000,d1
wb   mulu #100,d2
bV   mulu #10,d3
pp   add.l d0,d1
y0   add.l d2,d1
36   add.l d3,d1
8C   add.l d4,d1
ep   lea puffer,a0
1H   moveq #3,d3
1r   moveq #15,d4
750 he1:
jc1  rol.w #4,d1
jF   move d1,d2
ug   bsr nibble
S7   move.b d2,(a0)+
BI   dbra d3,he1
tR   move.l dosbase,a6
SU   move.l handle,d1
Qv   move.l #puffer,d2
El   move.l #pufferend-puffer-1,d3
DS   jsr write(a6)
BJ   bsr retout
0e   bra loop1
z70 are_you_sure:
1Z1  move.l dosbase,a6
ac   move.l handle,d1
u7   move.l #text10,d2
2a   move.l #text10end-text10,d3
La   jsr write(a6)
5C   moveq #80,d3
JB   bsr readchr
oI   cmp.b #"y",inline
GC   beq.s sure
xG   cmp.b #"n",inline
L2   beq.s notsure
fw   bra.s are_you_sure
qJ0 sure:
lm1  rts
3e0 notsure:
Hv1  bra loop1
520 headset:
g51  cmp.l #$00,head
7T   beq.s headset0
A3   clr.l head
lz   bra.s headsetend
cp0 headset0:
8R1  move.l #1,head
gc0 headsetend:
Tx1  bsr show1
R5   bra loop1
zA0 ;** Datendefinition **
1D   memende: dc.l 0
7f   memstart: dc.l 0
Wo   dezpuff: blk.b 5,32
6p   dezpuffend: dc.b 0
11   puffer:     blk.b 5,":"
Y8   pufferend:
81   byte:       blk.b 3," "
cu   byteend:
yx   char:       blk.b 16,0
gc   charend:
Ek   fbyte:      blk.b 5,0
f0   sbyte:      blk.b 5,0
2U   dbyte:      blk.b 3,0
G8   even
aM   from:    dc.l 0
1I   to:      dc.l 0
Au   with:    dc.b 0
KC   even
ro   adr:       dc.w 0
Y9   startadr: dc.w 0
uF   dosbase:  dc.l 0
c1   dosname:  dc.b "dos.library",0
GZ   fenstername: dc.b "CON:0/0/640/210/
     ** DiskSpy V1.0 **",0
QI   even
WC   handle: dc.l 0
vQ   titel: dc.b "  ** DiskSpy V1.0 **"
     ,$0a,13
eK   dc.b " 1987 by ",$9b,"1;3;31;40mMi
     chael Stein"
c9   dc.b $9b,"0;31;40m",$0a,$0a,13
Kj   titelende: dc.b 0
FX   helptext: dc.b "** Help-Page **",$
     0a
YM   dc.b "---------------",$0a
88   dc.b "l:List Helptext",$0a
AE   dc.b "x:Exit",$0a
SL   dc.b "d:Do sum",$0a
WQ   dc.b "h:Change Head",$0a
oj   dc.b "+:Next Cylinder",$0a
Sz   dc.b "-:Back Cylinder",$0a
04   dc.b "s:Show Cylinder,Sektor,Head a
     nd Block",$0a
Re   dc.b "c:Cli-Command,c again-get bac
     k to DiskSpy",$0a
xr   dc.b "r tt ss:Read Cylinder(0-79) S
     ektor(0-10)",$0a
7V   dc.b "w tt ss:Write Cylinder(0-79)
     Sektor(0-10)",$0a
Ix   dc.b "t tt:Show Cylinder(0-79)",$0a
sW   dc.b "f tt ll ",34,"text",34,":Find
     text between Cylinder"
Te   dc.b " tt and Cylinder ll",$0a
xB   dc.b "o mmmm nnnn dd:Fill buffer fr
     om mmmm to nnnn with dd",$0a
39   dc.b "m mmmm nnnn or m:Memorydump f
     rom mmmm to nnnn or from 0000"
j5   dc.b " to 01ff",$0a
94   dc.b "e mmmm:Edit from mmmm",$0a
7I   dc.b "?# or ?$:Compute to dez(0-655
     35) or to hex($0-ffff)",$0a,$00
KB   helptextend:
iC   text1: dc.b "Cylinder:",0
NP   text1end:
80   text2: dc.b "Sektor:",0
VT   text2end:
Ez   text3: dc.b "Searching for:",0
dX   text3end:
zk   text4: dc.b "** Found **!",0
lb   text4end:
8w   text5: dc.b "Block:",0
tf   text5end:
Fk   text6: dc.b "No Disk in Df0:!",0
1j   text6end:
bZ   text7: dc.b "Trackdisk-Device can'
     t be opened",0
9n   text7end:
UN   text8: dc.b "=$",0
Hr   text8end:
a0   text9: dc.b "=#",0
Pv   text9end:
m4   text10: dc.b "Are you sure (y/n) ?"
     ,0
4G   text10end:
rD   text11: dc.b "Head:",0
8L   text11end:
Gw   point: dc.b ".",0
7a   pointend:
oc   ret: dc.b $0a,0
6U   retend:
Gy   prompt: dc.b "1>",0
qx   promptend:
wj   was:   dc.b "Unknown Command!",$0a
     ,0
4S   wasend:
3P   trddevice: dc.b "trackdisk.device",
     0
ME   even
vv   rette:    dc.l 0
Yf   modus:    dc.w 0
du   track:    dc.l 0
S0   sektor:   dc.l 0
9m   head:     dc.l 0
ip   block:    dc.l 0
hs   endtrack: dc.l 0
Bn   inline:   blk.b 80,0
hi   locate:   blk.b 80,0
rN   locateend: dc.b 0
XP   even
VA   diskio:    blk.l 20,0
kn   readreply: blk.l 8,0
y9   diskbuff:  blk.b 512,0
(C) 1988 M&T
```

Listing.
»DiskSpy«, ein Diskettenmonitor, geschrieben mit dem Seka-Assembler (Schluß)

Funktionsliste

Viele Amiga-Benutzer wollen etwas tiefer in das Betriebssystem einsteigen, aber wissen nicht so recht, wo sie anfangen sollen. Unser Listing hilft Ihnen.

Der einfachste Zugang ist wohl der von AmigaBasic aus, wo man einzelne Funktionen schon mal ausprobieren kann, ohne gleich einen C-Compiler anschaffen zu müssen. Einige der in den Basic-Demos mitgelieferten Beispiele bieten dazu praktische Anwendungen von unterschiedlichem Schwierigkeitsgrad.

Dabei macht man auch Bekanntschaft mit den FD-Files, die irgendwo auf der ExtrasD-Diskette mitgekommen sind. Das heißt man arbeitet nicht mit den FD-Files selbst, sondern nur mit den aus ihnen abgeleiteten .bmap-Files.

Hier möchte ich aber jedem empfehlen, sich diese FD-Files einmal anzusehen, zum Beispiel mit dem TYPE-Befehl im CLI. Man stellt überrascht fest, daß es sich um lesbare Klartext-Files handelt, die die verfügbaren Systemfunktionen enthalten und noch zusätzliche Informationen über diese liefern.

Das Programm in Listing 1 erstellt nun aus diesen FD-Files eine Liste der verfügbaren Systemfunktionen, ordnet sie alphabetisch und schreibt noch dazu, in welcher System-Library sie jeweils zu finden sind. Die meisten Funktionen haben ja netterweise einen selbsterklärenden Namen. Man kann also in der Liste nach dem gesuchten Stichwort suchen und findet meist sehr schnell eine Funktion, die die gewünschte Aufgabe erfüllt. In der Liste findet man auch noch Angaben, welche Eingangsparameter die Funktion braucht. Beispielsweise den Pointer auf die RastPort-Struktur, die man sich ja per WINDOW(8)-Funktion im Amiga-Basic besorgen kann.

Praktisch arbeitet das Programm so, daß zunächst per Execute& die CLI-Funktion LIST aufgerufen wird, um eine Liste der überhaupt vorhandenen FD-Files zu erhalten. LIST hat den Vorteil gegenüber DIR, daß die Filenamen im Resultat immer am Zeilenanfang stehen und nicht wie bei DIR immer zu zweit nebeneinander. Weil hier die Execute&-Funktion benutzt wird, muß übrigens der Anfangs-CLI, der beim Booten ja aktiv ist, noch »am Leben« sein, sonst findet das AmigaDOS mit Ein- und Ausgabekanälen womöglich nicht mehr durch. Also bitte den abschließenden EndCLI-Befehl aus der Startup-Sequence entfernen (das machen Programmierer sowieso) oder diese vorzeitig mit <Ctrl-D> abbrechen.

Die vom LIST-Befehl erzeugte Datei namens fdlist wird im folgenden ausgewertet und nach Namen von FD-Dateien (und nur diesen) durchsucht. Wird eine gefunden, wird sie sofort geöffnet und ihrerseits durchgelesen. Der Name der Library wird aus dem Filenamen extrahiert, um ihn als a$ an alle Zeilen anhängen zu können. Kommentar- und sonstige hier uninteressante Zeilen werden ignoriert, die anderen in fu$() gesammelt. Nachdem so die Liste aller FD-Dateien durchgearbeitet ist, wird die Datei fdlist sofort wieder gelöscht, man soll ja keinen Müll hinterlassen, sondern immer sauber aufräumen.

Im folgenden Absatz werden die Einträge im Array fu$() sortiert. Im letzten Programmteil soll schließlich die Ausgabe der sortierten Liste erfolgen. Vorher wird man aber gefragt, ob das gleich auf den Drucker geschehen soll oder vorher bloß auf den Bildschirm oder dauerhaft in eine zu benennende Datei. Die endgültige Ausgabeschleife wirkt dann schon fast trivial.

Am Schluß sei noch zu bemerken, daß die so erstellte Funktionenliste beileibe nicht nur für Basic-Programmierer interessant ist. Auch der C- oder Assembler-Programmierer erhält hier eine ungeheuer praktische Übersicht, die ihm manches langwierige Wälzen in den ROM Kernel Manuals erspart oder doch zumindest stark verkürzt. (Dr. Peter Kittel/rb)

Programmname:	FktListe
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2
Bemerkung:	Initial-CLI muß noch offen sein

```
Programm : FktListe
-------------------

Q50 '" Programm  fktlist
jH  '" Dr. Peter Kittel, 14.1.88
UK  '" Erstellung einer Liste der verfügbaren
69  '" Betriebssystemfunktionen aus den FD-Files
gE  CLEAR ,50000&
jz  PRINT :PRINT "Dieses Programm funktioniert nur,
Ws  PRINT "wenn der Einschalt-CLI noch aktiv ist,
yB  PRINT "also evtl. die EndCLI-Anweisung in der
Gs  PRINT "Startup-Sequence lahmlegen!
cw  DIM fu$(2000) '" reichliche Schätzung
7b  DECLARE FUNCTION Execute& LIBRARY
Nk  LIBRARY "dos.library"
C3  PRINT :PRINT "Bitte Namen (incl.Zugriffspfad) der
79  PRINT "Schublade mit den fd-Files angeben
d5  PRINT "(normalerweise ExtrasD:fd1.2) :
tc  LINE INPUT di$
Un  x=Execute&(SADD("list >ram:fdlist "+di$+CHR$(0)),0,0)
wA  LIBRARY CLOSE
Fe  OPEN "ram:fdlist" FOR INPUT AS 1
m8  n=0:PRINT
mm  WHILE NOT EOF(1)
I42   LINE INPUT #1,a$
hJ    i=INSTR(a$,"_lib.fd")
XA    IF i>1 THEN
1E4     OPEN di$+"/"+LEFT$(a$,i+6) FOR INPUT AS 2
V2      a$=" /"+LEFT$(a$,i-1)
yu      WHILE NOT EOF(2)
hG6       LINE INPUT #2,b$:l$=LEFT$(b$,1)
QM        IF l$<>"#" AND l$<>"*" THEN n=n+1:fu$(n)=b$+a$:PRIN
          T fu$(n)
vj        WEND
M74     CLOSE 2
sV      PRINT
TM      END IF
zn2   WEND
N70 CLOSE 1
Ts  KILL "ram:fdlist"
ip  PRINT :PRINT "Sortieren nach Heapsort,";n;"Eintragungen"
uI  FOR HG=INT(n/2)+n-1 TO 1 STEP -1
bi2   HI=1:HR=HG:IF HG>=n THEN HI=HG-n+1:HR=n:HX$=fu$(HI)
fF    IF HR<n THEN HX$=fu$(HR+1):fu$(HR+1)=fu$(1):fu$(1)=HX$
Mb    IF HG<>1 THEN
Mr4     H1=-1
De      WHILE H1
bL6       HJ=2*HI
7o        IF HJ<HR THEN IF UCASE$(fu$(HJ))<UCASE$(fu$(HJ+1)) TH
          EN HJ=HJ+1
6N        H1=UCASE$(HX$)<UCASE$(fu$(HJ))
g9        IF H1 THEN fu$(HI)=fu$(HJ):HI=HJ:H1=2*HI<=HR
D1        WEND
Lp4     fu$(HI)=HX$
kd      END IF
uz2 . NEXT
zY0 PRINT :PRINT "Ausgabe auf Drucker(d), Datei(f) oder Schirm(s)
     ? ";
H2  a$="":WHILE a$<>"d" AND a$<>"f" AND a$<>"s":a$=INKEY$:W
    END
8F  PRINT a$:PRINT
wh  IF a$="s" THEN
JS2   OPEN "scrn:" FOR OUTPUT AS 1
bKA           ELSEIF a$="d" THEN
Qj2   OPEN "prt:"  FOR OUTPUT AS 1
YHO                          ELSE
FW2   PRINT :PRINT "Bitte Dateinamen (ggf. incl. Pfad) angeben:
C4    LINE INPUT a$:PRINT
CG    OPEN a$ FOR OUTPUT AS 1
xq    END IF
kz0 FOR i=1 TO n:PRINT #1,fu$(i):NEXT
rb  CLOSE 1
oj  END
(C) 1988 M&T
```

Listing 1. »FktListe« zeigt alle Funktionen an

Wahr oder falsch? Das ist hier die Frage!

Die boolesche Algebra ist nicht jedermanns Sache. Doch sind die Wahrheitstabellen für einige Anwendungen sehr wichtig. »Wahrtab« bestimmt diese nützlichen Tabellen in Sekunden.

Eine Anwendung von Wahrheitstabellen ist die Berechnung von logischen Schaltungen. Dabei sind die Eingangssignale durch verschiedene Logikfunktionen miteinander verbunden. Diese Funktionen lassen sich auch mit den Basic-Befehlen darstellen.

Bei dem Programm Wahrtab können Sie bis zu neun Eingangssignale (A — I) verwenden. Bei der Eingabe des Ausdrucks müssen Sie darauf achten, daß die Variablen groß und die Operatoren klein geschrieben werden. Also zum Beispiel:

```
A and B
```

Folgende Operatoren können Sie verwenden:

```
not, and, or, xor, eqv, imp
```

Diese möglichen Verknüpfungen finden Sie auf der Seite 8-16 Ihres Basic-Handbuchs genauer beschrieben. Dort sehen Sie auch Wahrheitstabellen, mit denen Sie gleich das Programm (siehe Listing) testen können. Mit Klammern können Sie natürlich auch die Reihenfolge der Abarbeitung bestimmen. Dies ist sehr wichtig, da zum Beispiel die zwei Ausdrücke

```
A and (B or C) und (A and B) or C
```

unterschiedliche Ergebnisse liefern.

Das Programm wandelt die eingegebenen Parameter (A, B, C ...) in die Feldvariablen a(0), a(1), a(2) und so weiter um. Wenn Sie unbedingt mehr als zehn Parameter benötigen, müssen Sie die FOR-Schleife in diesem Programmteil erweitern. Vergessen Sie aber nicht, daß dann die Feldvariable a mit dem DIM-Befehl vergrößert werden muß.

Dieser für Basic verständliche Programmteil wird nun in die RAM-Disk geschrieben und von dort mit dem Befehl CHAIN MERGE (Zeile 29) an das Ende des Programms angehängt. Dabei werden auch die Zeilen ab der Nummer 100 gelöscht. Deshalb müssen Sie die Zeilennummer 100 (im Listing Zeile 68) auch unbedingt miteingeben. Ansonsten funktioniert der MERGE-Befehl nicht. Dieses Vorgehen ist die einzige Methode, sich ändernde Ausdrücke in einem Basic-Programm zu verwenden, ohne im Listing selbst etwas einzugeben.

Die eigentliche Berechnung des Wahrheitsgehalts findet dann in diesem angehängten Programmteil statt. Sie wird von der Funktion »schachtel« (ab Zeile 48) aufgerufen. Hierbei wird durch den rekursiven Aufruf erreicht, daß immer jede mögliche Wahr-Falsch-Kombination berücksichtigt wird. Die Variable »tiefe« enthält dabei die augenblickliche Schachtelungstiefe.

Da die Wahrheitstabellen schnell sehr groß werden, ist es sinnvoll, die Tabelle auch auf dem Drucker auszugeben. Dazu tauschen Sie alle PRINT-Befehle, die die Ergebnisse auf dem Bildschirm ausgeben, durch einen LPRINT-Befehl aus. Das ist nötig in den Zeilen 37 bis 39, 52 und 56.

Wahrtab kann Ihnen also viel Arbeit abnehmen, oder wissen Sie auf Anhieb die Wahrheitstabelle für den folgenden Ausdruck?

```
A and (B eqv not C) or D
```

Wahrscheinlich nicht, und genau in solchen Fällen hilft Ihnen das Programm schnell und sicher weiter.

(Christian Zimmer/rb)

Programmname:	Wahrtab
Computer:	A500, A1000, A2000 mit Kickstart
Sprache:	Amiga-Basic 1.2

```
 1 L80 ' von Christian Zimmer
 2 6h  INPUT "Welcher Ausdruck";a$
 3 CT  zahl=0
 4 tn  ' Folgender Teil ersetzt Variablennamen (A,B,...) durch Feldv
       ariablen (a(0),a(1),...)
 5 uI  b$=a$
 6 Yx  FOR i=0 TO 9
 7 0x2   l=0
 8 4p    IF INSTR(b$,CHR$(65+i))>0 THEN
 9 Us4     loop:
10 SS      l=l+1
11 6h      IF l<=LEN(b$) THEN
12 oY6       IF MID$(b$,l,1)=CHR$(65+i) THEN
13 4s8         b$=LEFT$(b$,l-1)+"a("+RIGHT$(STR$(i),1)+")"+RIGHT$(b$
               ,LEN(b$)-l)
14 ia          l=l+3
15 B46       END IF
16 lo        GOTO loop
17 D64     END IF
18 tc2   ELSE
19 zB4     y=i-1
20 hk      i=9
21 HA2   END IF
22 RW0 NEXT
23 Jg  ' Hier wird der umgewandelte Ausdruck in das Programm eingefu
       egt
24 eb  OPEN "ram:Ausdruck" FOR OUTPUT AS 1
25 sK2   PRINT #1,"100 'Der logische Ausdruck"
26 Ec    PRINT #1,"  zz="+b$
27 BG    PRINT #1,"IF zahl=0 THEN GOTO wahrheit ELSE RETURN"
28 G00 CLOSE 1
29 Y8  CHAIN MERGE "ram:Ausdruck",100,ALL,DELETE 100-
30 wI  wahrheit:
31 gy2   zahl=1
32 sV    PRINT
33 pV    FOR tt=0 TO y
34 kD4     dr$=dr$+" "+CHR$(65+tt)+" "
35 ej2   NEXT
36 4E    dr$=dr$+"  "+a$
37 ry    PRINT dr$
38 wA    PRINT STRING$(LEN(dr$),"-")
39 zc    PRINT
40 Kb    tiefe=-1
41 gX    dr$=""
42 UY    GOSUB schachtel
43 3g    PRINT
44 kc    PRINT "Drücken Sie eine Taste"
45 YS    WHILE INKEY$="":WEND
46 tI    RUN
47 5q0 ' Nun folgt die eigentliche Berechnung der Tabelle
48 mS  schachtel: ' Ruft sich rekursiv selbst auf
49 LS2   IF tiefe=y THEN
50 L04     GOSUB 100
51 HT      FOR gg=0 TO tiefe
52 Tn6       IF a(gg)=0 THEN PRINT " f "; ELSE PRINT " w ";
53 w14     NEXT
54 r7      IF zz<0 THEN zz=zz+2
55 TC      PRINT SPACE$(INT(LEN(a$)/2));
56 es      IF zz=0 THEN PRINT " f" ELSE PRINT " w"
57 wS      tiefe=tiefe-1
58 Iu      RETURN
59 tm2   END IF
60 tN    tiefe=tiefe+1
61 Wb    a(tiefe)=0
62 os    GOSUB schachtel
63 wQ    tiefe=tiefe+1
64 fg    a(tiefe)=1
65 rv    GOSUB schachtel
66 5b    tiefe=tiefe-1
67 R30 RETURN
68 i7  100 'Der logische Ausdruck
69 DU2   zz=(a(0) AND a(1)) OR (a(2) AND a(3)) AND (a(4) OR a(5))
70 350 IF zahl=0 THEN GOTO wahrheit ELSE RETURN
(C) 1988 M&T
```

Listing. Mit »Wahrtab« wird das Berechnen von Wahrheitstabellen zum Kinderspiel

DAS AMIGA-GLOSSAR Teil 8

Die einzelnen Teile des Glossars zum Heraustrennen bilden ein umfassendes Nachschlagewerk der gängigsten Begriffe rund um den Amiga. Sammeln Sie mit.

Mit dieser Folge schließen wir unser Nachschlagewerk zum Amiga ab. War das Glossar, dessen erster Teil in der Ausgabe 8/9 '87 des AMIGA-Magazins erschien, ursprünglich als Hilfe bei der Arbeit mit der zu dieser Zeit größtenteils englischsprachigen Dokumentation konzipiert, ist es inzwischen zu einer wertvollen Informationsgrundlage für alle Anwender dieses wohl erstklassigen Computers geworden. Manchem Leser mögen die Ausführungen zu einzelnen Stichworten nicht ausführlich genug gewesen sein. Das ließ sich mit dem zur Verfügung stehenden Platz leider nicht vermeiden. Wenn Sie das eine oder andere Stichwort vermißt haben, schreiben Sie uns. Wir berücksichtigen Ihre Anregungen beim Verfassen einer nach deutschen Stichworten geordneten Sonderausgabe des Glossars.

(David Twigg-Flesner/pa)

startup-sequence (die Bootdatei) Eine besondere Systemdatei, die sich im Verzeichnis »s« befindet und eine Reihe von Befehlen enthält (batch file), mit denen das Betriebssystem des Amiga initialisiert wird. Amiga-DOS sucht diese Datei nach Einlegen der Bootdiskette (Systemdiskette) als erstes und führt die darin enthaltenen Befehle aus. Findet das DOS die Datei nicht, kann die Systemsteuerung nicht an die Workbench oder ein anderes Programm übergeben werden, sondern bleibt bei Amiga-DOS, das dann nur das CLI-Window initialisiert. Ein solches CLI darf nie mit ENDCLI ausgeschaltet werden, solange das System nicht mit einer anderen Form der Eingabesteuerung (ein zweites CLI oder LOADWB) »ansprechbar« ist. Sonst erfolgt ein -> lockup. Unter Version 1.2 ist die »startup-sequence« auch bei auto.config-Erweiterungen wichtig, weil diese erst mit den entsprechenden Befehlen dieser Datei, BINDDRIVERS beziehungsweise MOUNT, in das Betriebssystem eingebunden werden.

stdin, stdout, stderr Die vom System voreingestellten Standardgeräte für Ein-/Ausgabe und Fehlermeldungen (Tastatur,Bildschirm,Bildschirm) können vom CLI mit -> i/o redirection für die Ausführung eines Befehls »umgeleitet« werden.

stream (Datenstrom) Ein Datenübertragungskanal zwischen dem Computer und einem Peripheriegerät. Der Kanal wird durch ein -> file pointer (filing system) identifiziert, wenn die Daten nicht gepuffert werden; sonst durch ein -> file descriptor oder, unter Amiga-DOS, von einem -> file handle.

style (Schriftart) Die Zeichensätze des Amiga können in vier verschiedenen Schriftarten dargestellt werden: plain (der normale Zeichensatz), bold (Fettdruck am Bildschirm, Breitdruck bei den meisten Druckern), italic (Schrägdruck) und underlined (unterstrichen). Die nicht normalen Zeichensätze werden unter Verwendung von Systemalgorithmen abgeleitet. Die Ergebnisse dieser Berechnungen sind nicht immer optimal. Zeichensätze mit fest definierten Schriftarten werden von den Systemalgorithmen nicht modifiziert, wenn die entsprechenden Schalter (flags) in der Zeichendefinitionsdatei (.font) gesetzt sind.

ST506 Festplatten-Controller für maximal zwei Geräte mit mehreren Platten (bis zu 8 pro Festplatte). Der Controller befindet sich neben den -> SCSI auf der Festplatten-Karte zum Amiga 2000. Die Übertragungsgeschwindigkeit beträgt etwa 700 KByte/Sekunde.

sub-menu -> menu

SuperBitMap Ein -> Fenster (window) mit eigener Bitmap (siehe refresh). Diese Technik vereinfacht das Auffrischen des Fensterinhalts.

synchronous (synchron, gleichzeitig) Streng genommen bedeutet diese Bezeichnung, daß zwei oder mehrere Aktionen gleichzeitig stattfinden. Während die Autoren des -> RKM Hardware dieser Definition folgen, werden im RKM Exec nacheinander erfolgende Aktionen damit gekennzeichnet.

SYS: Dieses Verzeichnis ist gleichbedeutend mit der Startdiskette. In der Regel ist das die Workbench. Wann immer Dateien von der Workbench benötigt werden, sollte anstatt dem Namen des Laufwerks, in dem sich die Workbench (eventuell) befindet, SYS: angegeben werden. (siehe logical device)

SYSTEM: Dieses Verzeichnis enthält von Amiga-DOS oft benötigte -> tools. Darunter befindet sich das -> CLI sowie die Befehle Format, Diskcopy, ScreenDump (1.2), SetMap (1.2) und andere.

system-configuration Diese Datei enthält die mit -> Preferences eingestellten Systemparameter.

system disk Eine Diskette im Amiga-DOS-Format, die unabhängig von der -> Workbench bootfähig ist (siehe boot block). Auch als Bezeichnung für Diskette benutzt, mit der das System initialisiert wird. (siehe SYS:)

system log jam Ein »Verkehrsstau« an einem -> message port, der in der Regel dadurch verursacht wird, daß eingegangene messages nicht schnell genug bearbeitet beziehungsweise zurückgeschickt werden. Bei einigen message-Typen wird in solchen Fällen dem message port zusätzlicher Speicher zugewiesen. Eine Freigabe des Speichers erfolgt jedoch nicht bei der Abarbeitung der message, sondern erst mit Schließen des dazugehörigen Fensters. Das kann unter Umständen zu Speicherengpässen (out of memory) führen. Eine solche Situation kann eventuell mit Schließen der betroffenen Fenster gelöst werden.

system power supply Die Stromversorgung für am -> expansion bus angeschlossenen Peripheriegeräte (zum Beispiel Speichererweiterungen). Diese ist beim A 1000 ausreichend für etwa 2 bis 4 MByte zusätzliches RAM. Eine Überlastung der Stromversorgung kann die CPU zerstören.

system time Grundlage für die Berechnung des Datums durch Amiga-DOS. Das Betriebssystem berechnet alle Datumsangaben relativ zum 1.1.1978. Falls der Anwender das Datum nicht ausdrücklich neu setzt (zum Beispiel mit SETDATE), wird es der zuletzt auf der Bootdiskette angelegten Datei entnommen und auf »eine Sekunde« später eingestellt. Das Systemkalendarium ist auf insgesamt 100 Jahre (bis 31.12. 2077) ausgelegt.

Die aktuelle Systemzeit (current time), die Zeit in Sekunden und Mikrosekunden seit dem letzten Einschalten (power up) des Systems, wird vom -> input.device benutzt, um das Auftreten eines -> events zu bestimmen.

system timing frequencies Taktfrequenzen des Systems. Ausgangspunkt für den Systemtakt ist die amerikanische Bildschirmträgerfrequenz von 3,579545 MHz. Diese entspricht bei 60 »Bildern« in der Sekunde einer Zeilenaufbaudauer von etwa 63 Mikrosekunden.

Während dieser Zeit stehen 227,5 -> memory access cycles zur Verfügung, was eine Dauer von etwa 280 Nanosekunden pro cycle ergibt. Dieser Wert wird als »system clock cycle«, kurz »clock«, bezeichnet (siehe time slot allocation). Die 68000-CPU wird mit einer Frequenz von 7,15909 MHz — also doppelt so schnell wie die Bildträgerfrequenz — getaktet. Die Sonderchips arbeiten mit 14,31818 MHz wiederum doppelt so schnell wie die CPU.

system timing interval (auch system clock cycle) Der Systemuhrtakt von 280 Nanosekunden (siehe system timing frequencies). Dieser wird mit einem Periodenwert von 1 gleichgesetzt. Um den Periodenwert für die Tonerzeugung zu bestimmen, wird die Dauer einer einzelnen Stichprobe (sample) durch den Systemuhrtakt geteilt. Die höchstmögliche Stichprobenfrequenz liegt bei 28867, die kürzeste Stichprobendauer ist demnach 1/28867 Sekunde, was einem Periodenwert von 124 entspricht (siehe period). Bei niedrigeren Periodenwerten reicht die Zeit nicht mehr aus, um neue Daten einzulesen.

T: Das »temporary«-Verzeichnis (temporär: zeitweilig, zum Übergang). Hier legen unter anderem die CLI-Editorroutinen ED und EDIT Temporärdateien oder Sicherungskopien der zuletzt bearbeiteten Versionen von Texten ab. Unter Version 1.2 erscheint dieses Verzeichnis nicht notwendigerweise auf einer Diskette, weil es, anders als bei der Version 1.1, nur bei Bedarf vom jeweiligen Editor eingerichtet wird.

table look-up/TLU (Nachschlagetabelle) Ein Verfahren, um Datenwerte schnell umzuwandeln. Die »Zieldaten« werden tabellarisch in einer Datei abgelegt und können entweder über einen Feldindex oder durch Zeichenkettenvergleich gegen die Eingangsdaten ausgetauscht und damit umgewandelt werden.

task Ein lauffähiges Codemodul, bestehend aus Codeblock, -> Stack und -> task control block, welches nach Bedarf geladen oder gelöscht wird. Tasks sind die Grundlage eines -> process. Ein task wird erst nach der Zuweisung eines -> IDCMP und/oder message ports zu einem process.

task control block (TCB) Eine Exec-Datenstruktur, in der sich Informationen zu einer task befinden. Diese wird angelegt bei der Einordnung der task in einer -> queue (Warteschlange). Das -> include file »exec/task.h« beziehungsweise »exec/task.i« enthält die Definition dieser Datenstruktur.

task state Der aktuelle Status einer task. Man unterscheidet sieben Stufen: added (neu gestartet, noch nicht eingestuft), waiting (im »Schlafzustand«, wartet auf einen -> event), ready (wartet auf Zugang zu den -> resources), running (hat gerade Zugang zu den resources), exception (Fehler aufgetreten), removed (abgelaufen, befindet sich beim -> cleanup) und invalid (wird ignoriert).

terminal (wörtlich Endstation) Datenein/ausgabegerät. Meist besteht dieses aus Tastatur und Bildschirm (Datensichtgerät/VDU).

terminator (end-of-name) Namenabgrenzungszeichen. Beim Amiga wird normalerweise das Leerzeichen als terminator interpretiert. Aus diesem Grund müssen Dateinamen, die Leerstellen enthalten, mit doppelten Hochkommata (") eingeschlossen werden. Das geshiftete Leerzeichen (non-break space) dagegen wird nicht als terminator angesehen, es sind bei dessen Verwendung keine Hochkommata erforderlich.

tick Die Zeitmeßeinheit des Systems. Sie beträgt 1/50 Sekunde (siehe IntuiTick)

time division multiplexing (TDM) Das »Aufteilen« eines Übertragungskanals oder einer -> resource auf mehrere Routinen, die jeweils eine bestimmte Benutzungsfrist (-> quantum) zugewiesen bekommen. Die Verfahren von -> exec basieren auf einem ähnlichen Prinzip (-> round robin). Die Begriffe time-sharing/time-slicing sind Synonyme für TDM.

timer.device Subsystem, das für die Zeitmessung benutzt wird. Es stehen zwei Verfahren zur Verfügung. Das erste Verfahren richtet sich nach dem -> vertical blanking interval (UNIT_VBLANK) und wird für langfristige Messungen empfohlen, weil die Genauigkeit dann ziemlich konstant bei ± 1/60 Sekunde liegt. Das zweite Verfahren (UNIT_MICROHZ) benutzt einen programmierbaren Timer in einem der beiden 8520 CIA-Chips und ist empfehlenswert für kurzfristige Messungen. In beiden Fällen kann nur eine Mindestzeit und nicht eine genaue Zeit gemessen werden, weil die Zeitmessung vom Multitasking-Subsystem beeinträchtigt wird.

timer.library Systemroutinen für Zeitarithmetik

time slot allocation (DMA) Von den 227,5 Speicherzugriffszyklen (siehe system timing frequencies) stehen 226 den auf den Speicher zugreifenden Subsystemen zur Verfügung. Die Systeme greifen in der folgenden Reihenfolge zu:
memory refresh (Speicherzellenauffrischung) 4 Zyklen; Laufwerk-DMA 3 Zyklen; Audio-DMA 4 Zyklen (1 Zyklus pro Kanal; 2 Datenbyte pro Kanal/Zyklus; Sprite-DMA 16 Zyklen (2 Zyklen pro Spritekanal; ein Datenwort pro Zyklus); Bitplane-DMA 80 Zyklen (bei 4 Bitplanes; 20 Zyklen zu jeweils einem Datenwort); frei 10 Zyklen

Bei mehr als vier niedrigauflösenden Bitplanes müssen Zyklen, die eigentlich der CPU zustehen, benutzt werden. Im hochauflösenden Modus werden die letzten beiden Zyklen der Sprite-DMA verwendet, wodurch Sprite 7 nicht mehr benutzt werden kann.

Um Zeit für das Auslesen überbreiter -> raster (Overscan) zu haben, schaltet das System der Reihe nach die Sprite-DMA 7 bis 1 (je nach Breite des rasters) ab. Nur Sprite-Kanal 0, der für den Mauszeiger benötigt wird, ist nicht von dem erweiterten Anspruch der Bitplane-DMA betroffen.

title bar Die Titel- oder Namensleiste eines Windows. (siehe gadget/drag gadget)

TmpRas (temporary raster) Ein vorläufiger Arbeitsspeicher, der einem -> RastPort zugewiesen wird. In diesem Übergangsspeicher werden grafische Darstellungen »vorab« erzeugt, bevor sie in das -> raster selbst kopiert werden. In der Regel muß ein TmpRas denselben Umfang wie das raster haben.

toggle (Kipp-/Druckschalter) Solche Schaltfunktionen finden Anwendung in attributiven Menüfunktionen (zum Beispiel Schriftart bei Textverarbeitung) und Booleschen gadgets. Die Bezeichnung »toggle select« bedeutet, daß eine gewählte Funktion solange wirksam bleibt, bis diese durch Wahl einer anderen Funktion wieder abgestellt wird.

tool (Werkzeug) (Hilfs)Programm der -> Workbench.

track (Diskettenspur) Eine der kreisförmigen »Rillen« einer Diskette, die in eine systemspezifische Anzahl von -> sectors (auch blocks) eingeteilt wird. (siehe cylinder)

trackdisk.device Systemroutinen/Hardware für die Steuerung eines Laufwerks (disk controller) und die Übertragung von Daten zwischen -> main memory und secondary memory. Mit dieser Lösung kann der Amiga, bei entsprechender Programmierung des trackdisk.device, Disketten in verschiedenen Aufzeichnungsformaten verwalten (siehe pulse code modulation). Unter Version 1.2 werden Laufwerke im 3½- oder 5¼-Zoll-Format verwaltet, unter 1.1 nur im 3½-Zoll-Format. Für 5¼-Zoll-Laufwerke ohne automatische Diskettenwechselleitung gibt es unter 1.2 den CLI-Befehl DISKCHANGE.

Die Datenübertragung erfolgt als »track i/o«, das heißt, eine vollständige Spur wird von der Diskette in den Diskpuffer eingelesen beziehungsweise auf die Diskette geschrieben. »sector i/o« ist nur begrenzt möglich. Wegen des Diskpuffers, der über einen -> DMA-Kanal bedient wird, benötigt jedes angeschlossene Laufwerk etwa 17 bis 20 KByte im -> Chip RAM, damit die Daten vom -> Blitter verschoben werden können. Um die Zugriffszeiten zu verkürzen, kann man unter Version 1.2 mit dem Befehl ADDBUFFERS den Umfang des Diskpuffers erhöhen — die Zahl hinter dem Befehl gibt die Anzahl der zusätzlichen Speicherblöcke zu je 512 Byte an. Die zusätzliche Speicherzuweisung läßt sich nicht rückgängig machen (es sei denn, man hat ein Laufwerk mit Ausschaltknopf).

Das Trackdisk.device kann bis zu vier Laufwerke (unter 1.2 in beliebiger Mischung aus 3½ und 5¼ Zoll) verwalten. Davon sollen laut Hersteller nur zwei vom Amiga selbst mit Strom versorgt

werden. Es wird auch davor gewarnt, 5¼-Zoll-Laufwerke ohne eigene Stromversorgung am -> drive port anzuschließen, weil viele dieser Geräte beim Einschalten eine zu hohe Stromaufnahme haben.

Transformer (der Umwandler) Die Software-Lösung der IBM-Kompatibilität beim Amiga. Das Programm läuft in der vorliegenden Version nur unter 1.1. Mit dem Erscheinen von -> Sidecar ist die weitere Entwicklung für die Version 1.2 eingestellt worden.

Nur Programme, die weder auf eine IBM-Grafikkarte (CGA, EGA, Hercules) noch auf das System-ROM des IBM-Computers zugreifen, können unter Transformer laufen. In anderen Fällen muß man Sidecar benutzen. Da IBM-Programme unter Transformer auch etwas langsam laufen, gibt es ein Public Domain-Programm namens Speedy3 für einen schnelleren Ablauf.

translator.library Die Systemroutinen für die Umwandlung von Zeichenketten in Phonemcodeketten. (siehe speech synthesis)

trap (Falle) Eine Art -> exception, die als Folge einer Operation der CPU entsteht. Anders als bei einer exception muß eine task auf eine trap-Meldung sofort reagieren, sonst wird die Ausführung des tasks abgebrochen (»task held«). Viele traps ergeben sich aus CPU-Fehlerzuständen, wie zum Beispiel Division durch Null, Adressenfehler und so weiter. Aus diesem Grund werden traps im »supervisor mode« verarbeitet, das heißt, der Systemstapelspeicher (supervisor stack) nimmt die gesicherten Registerinhalte auf. Diesen wird eine »longword exception number« hinzugefügt (siehe guru meditation).

Trashcan (Abfalleimer/Papierkorb) Eine besondere »Schublade« der -> Workbench. In dieser Schublade abgelegte Icons stellen Dateien dar, die zum »Wegwerfen« bestimmt sind. Die im Trashcan abgelegten Dateien bleiben solange erhalten, bis die Menüfunktionen »empty trash« (Müll rausbringen) gewählt wird. Danach sind sie gelöscht. Anders als andere Schubladen kann Trashcan weder in eine andere Schublade verlegt noch — selbst im völlig leeren Zustand oder mit einem anderen Namen versehen — vom Anwender gelöscht werden. In der Regel wird diese Schublade beim Formatieren einer Diskette automatisch auf der Diskette erzeugt. Seit Version 1.2 verfügt der CLI-Befehl FORMAT über eine »NOICONS«-Funktion, welche die Erzeugung eines Trashcans verhindert.

troubleshooting (Fehlersuche) Die Suche nach Hinweisen auf mögliche Fehlerquellen und Lösungswege. Die (gedankliche) Vorstufe zum »debugging«, der gezielten Fehlerbehebung.

tutorial (Kurs) Anders als »Kurse« in deutschen Computerzeitschriften, die sich in der Regel über mehrere Folgen erstrecken, ist das angelsächsische »tutorial« üblicherweise ein einziger, in sich abgeschlossener und Wissen vermittelnder Artikel (oder Kapitel eines Handbuchs) zu einem einzigen Aspekt der Computerei.

UART (universal asynchronous receiver/transmitter) Ein Baustein, der die Verbindung mit Peripheriegeräten über den seriellen Port steuert (siehe Paula/Portia).

unary minus Vorzeichenwechsel

user interface -> operating system

UserPort -> IDCMP eines Window

UTILITIES Schublade der -> Workbench. Sie enthält -> tools, welche die »Schreibtisch«-Funktion der Workbench unterstützen (-> Notepad und Calculator).

utility program (Hilfsprogramm). Ein (meist relativ kurzes) Programm, das die Bedienung oder Programmierung des Computers in einem bestimmten Aspekt verbessert (erweitert, Fehler behebt), wie zum Beispiel -> spooler, Menüeditor, Texteditor, -> shell und so weiter.

VDU (visual display unit) Datensichtgerät, Monitor oder Bildschirm.

vector graphic display Eine Bildschirmdarstellung, die aus Linien aufgebaut wird. Gegensatz: -> raster graphic, die zeilenweise aufgebaut wird. Der Unterschied zwischen den beiden Verfahren läßt sich am einfachsten mit dem Vergleich Matrixdrucker (Punkte, zeilenweise = raster graphic) und Plotter (Linien von unterschiedlicher Länge = vector graphic) verdeutlichen.

Version Die jeweilige Fassung eines Programms oder eines Betriebssystems. Unter 1.1 ist diese Information nur über Menüoption abrufbar, unter 1.2 gibt es auch einen entsprechenden CLI-Befehl. In frühen Meldungen aus Amerika zum Entwicklungsstand des Computers war von Kompatibilitätsproblemen zwischen Version 24 und 27 die Rede. Die Ursache dafür war vermutlich der Wechsel vom ursprünglichen Betriebssystem zu Amiga-DOS. Version 28 und 29 wurden in den Vorserienmaschinen benutzt, die den amerikanischen Software-Entwicklern zur Verfügung gestellt wurden. Version 30 wurde mit den ersten amerikanischen Geräten ausgeliefert (Version 1.0); Version 31 (Version 1.1) erschien zur europäischen Einführung des Amiga und Version 33 (1.2) ist die aktuelle ROM-Version des A 500 und A2000.

Beim Programmieren kann die Versionsnummer insbesondere einer -> library (Programmbibliothek) wichtig sein, denn beim Öffnen einer library wird eine Versionsnummer erwartet. Ist diese Angabe ungleich Null (eine beliebige Version der library wird akzeptiert), kann die library nur dann erfolgreich geöffnet werden, wenn die aktuelle Versionsnummer entweder mit der angegebenen identisch oder höher ist.

version.library Die Systemroutinen für die Verarbeitung von Version-Aufrufen

vertical blanking interval (auch vertical retrace) Die Zeit, in der der Elektronenstrahl keine Daten am Bildschirm darstellen kann, weil er sich gerade oben beziehungsweise unten am Bildschirm außerhalb des -> display field befindet. Die Mindestdauer dieser Darstellungsunterbrechung beträgt 20 Bildschirmzeilen (etwa 1/560 Sekunde), die normalerweise ab der obersten Zeile des Bildschirms berechnet wird. Während dieser Zeit läuft eine Unterbrechnungsroutine (interrupt), in der das System verschiedene Systemoperationen (vertical blanking tasks, wie Auffrischung der Darstellung ohne die Gefahr, daß Daten während der Phase des Bildaufbaus plötzlich verändert werden) ausführt. Die Höchstdauer beträgt die Anzahl »freier« Bildschirmzeilen nach Abzug des verwendeten display field (200/256 Zeilen). (siehe horizontal blanking, overscan)

video persistence Die Leuchtdauer der Phosphorbeschichtung eines Bildpunkts am Bildschirm, der vom Elektronenstrahl in den Farben Rot, Grün, Blau zum Leuchten gebracht wird. Je länger die Leuchtdauer, um so stabiler ist das erzeugte Bild. Um -> hires flicker im -> Interlace- Modus zu vermeiden, wird die Benutzung eines »high persistence«-Monitors mit lang nachleuchtender Beschichtung empfohlen.

View Die Bilddarstellung des Amiga besteht aus einem oder mehreren View-Ports (-> ViewPort). Die Datenstruktur View enthält unter anderem die Adresse des ersten View-Ports und der beiden für die Darstellung benötigten Copperlisten.

ViewPort Ein View-Port ist ein mit eigenen Bildattributen ausgestatteter, rechteckiger Darstellungsbereich auf dem Bildschirm. Es bildet den Hintergrund, auf dem Windows erzeugt werden und damit die Grundlage der Bildschirmverwaltung unter Intuition (siehe layer). ViewPorts lassen sich nicht nebeneinander positionieren, sondern müssen übereinander mit jeweils einer Bildschirmzeile Abstand angeordnet werden. Die Datenstruktur ViewPort enthält die Bildattribute: Farbkombination, Auflösung, Höhe und Breite, Koordinaten . . .

virtual. . . (scheinbar) Gegenstand, der die Attribute und Funktionen eines anderen Gegenstandes emuliert.

— virtual machine: ein »Scheincomputer«. Da beim Multitasking ein -> process — zumindest zeitweilig — die volle Kontrolle über alle Systeme des Computers hat, kann ein Multitasking-System aus mehreren virtuellen Computern bestehen.
— virtual processor: die Funktion einer task innerhalb eines process
— virtual screen: ein »Scheinbildschirm«. Ein Screen von Intuition
— virtual terminal: ein »Scheinterminal«. Obwohl sich Fenster nur einen Bildschirm und eine Tastatur teilen, verhält sich ein Fenster wie ein Terminal mit eigenem Ein- und Ausgabegerät
— virtual sprite (VSprite): -> GELs

VLSI (very large scale integration) Ein Baustein, der mehr als 100.000 Transistoren enthält.

volume (wörtlich Band; zum Beispiel eines Lexikons) Eine Diskette des Amiga.

voice (Stimme) Ein Audiokanal des Amiga.

Wack Ein »debugger«, also ein Hilfsprogramm, mit dem man Fehler in einem abgestürzten Programm finden und eventuell beheben kann. Eine Fassung mit reduziertem Befehlssatz ist als ROM-Wack im -> Kernel enthalten und bildet die debug()-Funktion von -> exec.

Um das Programm benutzen zu können, ist der Anschluß eines auf 9600 Baud Übertragungsgeschwindigkeit eingestellten Terminals am seriellen Port des Amiga notwendig. Bei einigen der älteren Programme kann man nach Erscheinen einer -> guru meditation nr. das ROM-Wack durch Klicken der rechten Maustaste aktivieren. Eine andere Methode ist das Betätigen der DEL-Taste am externen Terminal. Der Befehlssatz des Wack wird im -> RKM Exec ausführlich beschrieben.

Es handelt sich bei Wack um ein »memory editor program«, das heißt, die Daten werden direkt im Speicher des Computers geändert. Alle Adressensprünge erfolgen relativ zu einer gegebenen Basisadresse (in der Regel die Anfangsadresse der abgestürzten Routine).

Noch nicht abgestürzte Programme lassen sich mit ROM-Wack »überwachen«. Dazu muß die Anweisung LOADWB in der -> startup-sequence in »LOADWB -debug« geändert werden. In der Menüleiste der Workbench erscheint dann rechts neben »Special« ein weiteres Menü, dessen einziger Unterpunkt ROM-Wack startet.

wait list (die »Schlafschlange«) Die Exec-Liste der tasks, die auf ein -> event warten, um wieder aktiv zu werden. (siehe task state)

warm boot -> reset, reboot

window (Fenster) Das Grundelement der Bildschirmdarstellung unter Intuition. Die Schnittstelle zwischen Programmcode und dem Benutzer, also die optische Darstellung eines -> virtual terminal. Ein Fenster erscheint immer relativ zum -> origin eines Screens und kann innerhalb desselben frei bewegt, positioniert, in der Größe verändert und hinter oder vor anderen Windows »gestapelt« werden. (siehe gadget)

Der Amiga unterstützt vier verschiedene Fenstertypen:
— backdrop: erscheint grundsätzlich hinter allen anderen Windows (back: Hintergrund)
— borderless: besitzt keine »Einrahmung«; kann in Verbindung mit »backdrop« eine Art Screen simulieren
— GimmeZeroZero: ein Fenster mit getrenntem Speicher für die Einrahmung. Im Gegensatz zu anderen Fenstertypen hat dieses die Koordinaten 0,0 (Zero,Zero) tatsächlich in der linken oberen Ecke des Darstellungsbereiches und nicht in der Ecke unterhalb der Einrahmung.
— SuperBitMap: ein Fenster mit eigenem Speicher für den Darstellungsbereich (siehe refresh)

Die verschiedenen Windowtypen lassen sich auch kombinieren.

window pointer -> mouse pointer

WindowPort -> IDCMP

Windows Amiga-Basic Fenster Nummer 1 ist das Ausgabefenster von Amiga-Basic und daher nicht immer unter voller Programmkontrolle, weil der Benutzer Größe und Position ändern kann. In diesem Fenster erscheinen auch die Fehlermeldungen von Amiga-Basic und nicht in vom Programm definierten Fenstern.

Wegen der Fenstereinrahmung können von Amiga-Basic aus definierte Fenster nicht den vollen Bildschirmumfang einnehmen. Die Maximalgrößen betragen bei niedriger Auflösung:
0,0 - 311,186 ohne size gadget
0,0 - 297,186 mit size gadget
mittlerer Auflösung:
0,0 - 631,186 ohne size gadget
0,0 - 617,186 mit size gadget

Im Interlace-Modus erhöht sich die Grenze 186 auf 386. Bei den PAL-Amigas betragen die maximalen y-Koordinaten 242 und 498.

word alignment Ein »word« ist ein 16 Bit breiter Datenwert (2 Byte). Für verschiedene Operationen werden Daten benötigt, die an der Grenze eines solchen Wertes beginnen, also an einer genau durch zwei teilbaren Speicheradresse. Werte, die auf diese Weise im Speicher abgelegt werden, bezeichnet man als »word aligned«. (siehe longword)

Workbench (wörtlich »die Werkbank«; meistens »Arbeitstisch« oder »Schreibtisch«) Konsequent werden die Programme, mit denen der Computer arbeitet, Werkzeuge (Tools) genannt. Die Workbench selbst hat zwei Aufgaben:
1. Als »Werkbank« der Hauptbildschirm (screen) von Intuition«; eine »Arbeitsfläche« für die Programme.
2. Als »Werkzeug« selbst ein Programm, mit dem sich Dateien aus dem -> filing system von Amiga-DOS bearbeiten lassen. In dieser Eigenschaft gilt die Workbench neben Amiga-DOS als Ausgangspunkt für die Erzeugung von -> processes und ist somit auch ein wesentlicher und zugleich komfortabler Bestandteil des Multitaskingsystems.

Workbench disk Ein Mehrzweckbegriff: Eine Diskette mit Programmen, die -> Icons haben und mit der Maus gestartet werden können, eine bootfähige Diskette (siehe system disk) oder eine Programmdiskette, die einen Workbench-Screen benutzt.

workspace (Arbeitsspeicher) Ein Speicherbereich, in dem eine Routine oder ein Codesegment (meist lokale) Datenwerte ablegen kann. Ein -> hunk_bss in einem Codesegment erfüllt diese Funktion ebenso wie ein -> stack frame.

writable control store (WCS; auch write-once memory (WOM) oder Kickstart-RAM). Anstelle von ROM-Chips hat der A 1000 einen besonderen RAM-Speicher (siehe memory map), in welchen das Betriebssystem (-> Kernel) von der Kickstart-Diskette gelesen wird. Nach Übernahme des Kernels sperrt das Betriebssystem diesen Speicher für Schreibzugriffe und damit arbeitet er wie ein ROM-Speicher.

x/y coordinates Die Position eines Punktes am Bildschirm relativ zum »Ursprung« (origin), der in der linken oberen Ecke des Bildes liegt. Die y-Koordinate bestimmt die Zeile und die x-Koordinate den Punkt in der Zeile. Bei -> Sprites hat die y-Koordinate bei der Feststellung von Kollisionen Vorrang vor der x-Koordinate.

zero crossover Der »Nullpunkt« in einer Wellenform. Nur an dieser Stelle lassen sich zwei verschiedene Wellenformen ohne weiteres miteinander verbinden, ohne daß dabei klickende Störgeräusche entstehen.

zorro Die von Commodore-Amiga entwickelte Steckerbelegungsvorschrift für selbsteinbindende Erweiterungskarten (siehe auto.config). Anders als der -> expansion bus, der eine 86polige Belegung hat, schreibt die Zorro-Vorschrift einen 100poligen Stecker vor. Solche Karten werden in der Regel über ein -> expansion chassis am Computer angeschlossen. Leider gibt es für den Amiga zwei unterschiedliche Belegungen. Bei den rechteckigen Karten des Amiga 2000 sind sieben Pins anders belegt als bei den quadratischen Karten des Amiga 1000. Beide Vorschriften verlangen, daß mehrere Karten nach dem -> daisy chain-Prinzip zusamengeschaltet werden können.

Farben ändern wie ein Chamäleon

Die Schieberegler zur Farbeinstellung kennt wohl jeder zum Beispiel von professionellen Programmen her. Aber auch von Basic aus ist es möglich, so die gewünschten Farben zu erhalten.

Basic ist bislang auch auf dem Amiga die für den Hobby-Programmierer freundlichste Sprache: Sie wird am ehesten beherrscht und steht jedem Anwender zur Verfügung, weil sie im Lieferumfang enthalten ist. Und wenn die Möglichkeiten des Basic voll ausgereizt werden, dann lassen sich durchaus Programme gestalten, die die Fähigkeiten des Amiga gut nutzen. Denn das Amiga-Basic ist längst nicht so schlecht, wie sein Ruf.

Manchmal allerdings ist der Umgang mit der Hardware von Basic aus etwas beschwerlich. So ist zum Beispiel die Nutzung der zur Verfügung stehenden Farbenvielfalt durchaus nicht ganz einfach. Zwar läßt sich mit dem Befehl PALETTE jede Farbe nach eigenen Wünschen definieren. Wer es aber schon einmal versucht hat, die für einen gewünschten Farbton erforderlichen drei Zahlenwerte zwischen 0 und 1 experimentell herauszufinden, der weiß, wie zeitraubend das ist. Und wenn die mühsam bestimmten Farbwerte dann am nächsten Tag — bei anderem Licht — doch nicht so recht gefallen, dann spätestens macht man sich Gedanken über einen in das Programm integrierten Farbeinsteller, mit dem die Farben unabhängig von den in den »PREFERENCES« voreingestellten Farben im Programmlauf nach Lust, Licht und Laune verändert werden können. Professionelle Programme (wie beispielsweise Deluxe Paint) zeigen, wie angenehm eine solche Option ist. So etwas muß doch auch von Basic aus zu machen sein. Wie das geht, sehen Sie gleich.

Einfach und schnell

Ein paar Hürden sind allerdings zu nehmen, und die für eine komfortable Farbeinstellung notwendige Routine ist so umfangreich, daß es sich lohnt, sie als universell einsetzbares Unterprogramm zu gestalten, damit sie dann bei Bedarf an jedes Basic-Programm angehängt werden kann. Das Ergebnis solcher Überlegungen ist »ColorChange«. Dieses Programm ist also zur Bereicherung Ihrer Unterprogrammsammlung gedacht. Es ist nicht allein lauffähig, sondern wird an ein beliebiges Basic-Programm angehängt und verleiht dann diesem Programm den bequemen Zugriff auf die ganze Farbenpracht des Amiga.

Der universelle Einsatz von ColorChange wird durch die automatische Anpassung des Unterprogrammes an Größe und Farbtiefe des im Hauptprogramm aktivierten Screens möglich. Alle Variablen sind im Unterprogramm als lokale Variable definiert, so daß Konflikte mit den im Hauptprogramm verwendeten Variablen unmöglich sind. Trotzdem soll auf die verwendeten Variablen hier kurz eingegangen und dabei die Funktionsweise des Programms (siehe Listing) erläutert werden:

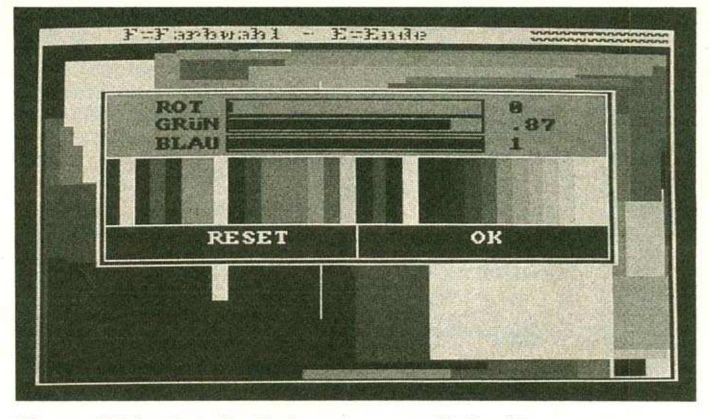
ColorChange ermittelt die Breite und Höhe des aktuellen Screens, legt abhängig hiervon die Maße für den aufzubauenden Farbrequester (feldx,feldy,bahn) fest und öffnet ein Window in entsprechenden Maßen (siehe Bild). Und nun wird es interessant:

Um die bisher eingestellten Farbwerte nötigenfalls später wieder rekonstruieren zu können, müssen diese Werte gelesen und zwischengespeichert werden. Dazu aber sind einige Hindernisse zu überwinden. Da ja bekanntlich der Amiga seinen Speicher dynamisch verwaltet, muß man sich mit etlichen PEEKs zunächst durch einen Zeigerwald hangeln, ehe man den Anfang der Farbtabelle gefunden hat. Hier sind nun die im aktuellen Screen gültigen Farbwerte mit jeweils zwei Byte für jede Farbe eingetragen. Liest man nun durch PEEK beispielsweise für den von der Workbench voreingestellten Screen die ersten vier Eintragungen in der Farbtabelle, so erhält man die rätselhaften Dezimalzahlen 90, 4095, 2, 3968. Erst die Umsetzung in hexadezimale Schreibweise läßt erkennen, wie in diesen Werten die Farbeinstellungen codiert sind:

```
005A (Farbe 0 = Workbench-Blau)
0FFF (Farbe 1 = Weiß)
0002 (Farbe 2 = Schwarz)
0F80 (Farbe 3 = Rot-Orange)
```

Diese Werte ergeben sich natürlich nur, wenn Sie nicht schon mit PREFERENCES die Farben geändert haben. Während die höchstwertige Stelle offenbar für die Farbcodierung nicht benutzt ist, enthalten die drei weiteren Stellen von links nach rechts jeweils den Helligkeitswert für die Grundfarbe Rot, Grün und Blau. Der Anteil jeder der drei RGB-Farben wird dabei hardwaremäßig jeweils von 4 Bit repräsentiert. Damit sind also 15 verschiedene Werte möglich. Aus der Kombination der 16 x 16 x 16 verschiedenen Farbanteile ergeben sich letztendlich die 4096 Farben, die der Amiga darstellen kann.

Mit diesen Erkenntnissen gewappnet können wir nun daran gehen, gezielt die Eintragungen in der Farbtabelle zu verändern. Nach der Zahl der im Screen zur Verfügung stehenden Farben (»farbzahl«) legen wir zuerst ein zweidimensionales Feld (»farbwert«) an, in welchem für jede im Screen verwendete Farbe vier Zahlenwerte gespeichert werden können. An erster Stelle wird der verschlüsselte Wert jeder Farbe so zwischengespeichert, wie er bei Aufruf der Subroutine eingestellt war. Anschließend wird aus diesen codierten Werten für jede Farbe der Rot-, Grün- und Blau-Anteil ermittelt und ebenfalls im Feld farbwert zwischengespeichert. Für das Hintergrund-Blau der Workbench würden sich also zum Beispiel folgende Eintragungen ergeben:



Dieses Bild zeigt die Farbregler von »ColorChange« vor dem Hintergrund des Demoprogramms

```
farbwert (0,0): 90 (codierter Wert)
farbwert (0,1):  0 (Rotanteil)
farbwert (0,2):  5 (Grünanteil)
farbwert (0,3): 10 (Blauanteil)
```

Nachdem für jede im Screen verwendete Farbe in dieser Weise die vorgefundenen Werte ermittelt und zwischengespeichert sind, kann nun der Farbrequester aufgebaut werden. Dabei dienen die Variablen »px%« und »py%« in Verbindung mit der Subroutine »curpos« der pixelgenauen Positionierung des Grafik- und Textcursors.

Der weitere Programmlauf erfolgt mausgesteuert über die Routine »mcheck«. Mit der Maus wird die jeweils gerade zu bearbeitende Farbe (»aktivfarbe«) ausgewählt und die mit der Maus eingestellten Reglerpositionen der Grundfarben (»posx«) werden in

die entsprechenden Helligkeitswerte umgerechnet und in die Feldvariable farbwert eingetragen.

Durch Anklicken des Feldes »RESET« wird zurückverzweigt zur Routine »rescol«, die die zwischengespeicherten ursprünglichen Farben reaktiviert. Erst ein Klick auf »OK« schließt die Farbwahl ab. Dabei werden die Inhalte des Variablenfeldes farbwert als Datei unter dem Namen »Farbdaten« auf Diskette gespeichert und stehen von nun an für die farbliche Gestaltung des Hauptprogrammes zur Verfügung.

Ein Beispiel für den Einsatz des Unterprogramms ColorChange gibt das kleine Demoprogramm, das nichts weiter bewirkt, als eine zufallsgesteuerte Farbspielerei auf dem Bildschirm. Es zeigt, wie vom Hauptprogramm die auf Diskette gespeicherten Farbdaten eingelesen und zur Farbeinstellung verwendet werden. Natürlich funktioniert das nur, wenn eine solche Datei auf der Diskette bereits existiert. Darum muß beim ersten Programmlauf die Leseroutine übersprungen werden. Wenn dann erst einmal das Farbdaten-File angelegt ist, kann und muß der Sprungbefehl »GOTO main« in Zeile 96 gelöscht werden.

Mit dem Demoprogramm können Sie nun die Arbeitsweise des Farbeinstellers nach Herzenslust testen und sich an den eigenen Farbkreationen freuen. Wenn Sie die in den beiden ersten Zeilen des Demoprogramms voreingestellten Werte für Farbzahl und Modus in den zulässigen Bereichen ändern, können Sie sich davon überzeugen, daß ColorChange seinen Dienst unabhängig von den gewählten Screenparametern errichtet und also — wie versprochen — jedes Basic-Programm um den Komfort freier Farbwahl zu verbessern hilft. (Jürgen Curdt/rb)

Programmname:	ColorChange
Computer:	A500, A1000, A2000 mit Kickstart 1.2
Sprache:	Amiga-Basic 1.2

```
  1 eR0 ' ** COLORCHANGE von Juergen Curdt **
  2 2q  SUB farbwahl STATIC
  3 Wi2   feldx=WINDOW(2)/10: feldy=WINDOW(3)/5: bahn=feldy/7
  4 wx    WINDOW 4,,(feldx,feldy)-(9*feldx,3.25*feldy+1),0,1
  5 qm    farbzahl=WINDOW(6)
  6 Nt    IF dimflag=0 THEN DIM farbwert(farbzahl,3): dimflag=1
  7 dE    Tabellenanfang=PEEKL(4+PEEKL(4+(44+PEEKL(WINDOW(7)+46))))
  8 o0    FOR i = 0 TO farbzahl
  9 WS4     farbwert(i,0)= PEEKW(Tabellenanfang+2*i)
 10 br2   NEXT i
 11 4d0 Farbspeicher:
 12 x22   aktivfarbe=0
 13 yG    COLOR 1,0
 14 mb    py%=2*(feldy+bahn): px%=2*(feldx-bahn): GOSUB curpos: PRIN
          T "RESET"
 15 RM    px%=6*feldx-bahn: GOSUB curpos: PRINT "OK"
 16 mK    FOR i = 0 TO farbzahl: hd=farbwert(i,0)
 17 aq4     FOR j=1 TO 3
 18 jD6       hx=hd\16^(3-j): hd=hd-hx*16^(3-j)
 19 XB        farbwert(i,j)=INT(hx*6.667)/100
 20 n44     NEXT j
 21 17      PALETTE i,farbwert(i,1),farbwert(i,2),farbwert(i,3)
 22 UP      LINE (i*8*feldx/(farbzahl+1),feldy)-((i+1)*8*feldx/(farb
            zahl+1),2*feldy),i,bf
 23 o42   NEXT i
 24 2y    LINE (0,0)-(8*feldx,2.5*feldy),1,b
 25 UE    LINE (0,feldy)-(8*feldx,2*feldy),1,b
 26 Uy    LINE (4*feldx,2*feldy)-(4*feldx,2.5*feldy)
 27 9Z    GOSUB reglerfeld
 28 YS0 mcheck:
 29 tk2   flag= MOUSE(0)
 30 yj    IF flag>0 THEN GOSUB farbfeldwahl
 31 R9    IF flag<0 THEN GOSUB regeln
 32 2V0 GOTO mcheck
 33 nG  regeln:
 34 cg2   IF MOUSE (2)>feldy-bahn OR MOUSE(2) <bahn THEN RETURN
 35 9t    i=(MOUSE(2)-6)\(bahn*2)+1
 36 kW    posx=MOUSE(1)
 37 Fu    IF posx<2*feldx OR posx>6*feldx THEN RETURN
 38 ft    py%=bahn*i*2: GOSUB regler1
 39 zb0 RETURN
 40 iD  farbfeldwahl:
 41 Oi2   y= MOUSE (4): IF y<feldy OR y>2.5*feldy THEN RETURN
 42 Oo    x=MOUSE(3): IF x<0 OR x> feldx*8 THEN RETURN
 43 E3    IF y<2*feldy THEN
 44 r44     aktivfarbe=INT(x/(8*feldx/(farbzahl+1)))
 45 Uz      GOSUB reglerfeld: RETURN
 46 gZ2   END IF
 47 eh    IF x<4*feldx THEN Farbspeicher
 48 Y80 'Farbwerte auf Diskette speichern:
 49 1C2   WINDOW CLOSE 4: MOUSE ON
 50 2z    OPEN "Farbdaten" FOR OUTPUT AS 1
 51 7f    PRINT#1,farbzahl
 52 Uc    FOR i=0 TO farbzahl
 53 AQ4     FOR j=1 TO 3
 54 2x6       PRINT#1,farbwert(i,j)
 55 eI4     NEXT j,i
 56 iS2   CLOSE 1
 57 4I    EXIT SUB
 58 Iu0 RETURN
 59 MX  reglerfeld:
 60 Ax2   LINE (0,0)-(8*feldx,feldy),aktivfarbe,bf
 61 iV    LINE (0,0)-(8*feldx,feldy),1,b
 62 TJ    zfarbe=farbzahl-aktivfarbe
 63 hX    COLOR zfarbe,aktivfarbe
 64 Iv    RESTORE reglerfeld: DATA ROT,GRÜN,BLAU
 65 BT    px%=feldx-bahn
 66 7E    FOR i = 1 TO 3
 67 Rs4     py%=bahn*i*2: GOSUB curpos: READ m$: PRINT m$
 68 EZ      posx=farbwert(aktivfarbe,i)*feldx*4+2*feldx
 69 Md      GOSUB regler2
 70 Rq      LINE (feldx*2-1,py%+2)-(feldx*6+1,py%-bahn-2),zfarbe,b
 71 EJ2   NEXT
 72 W80 RETURN
 73 h4  regler1:
 74 Hj2   farbwert(aktivfarbe,i)=INT((posx-feldx*2)/feldx*25)/100
 75 gK    PALETTE aktivfarbe,farbwert(aktivfarbe,1),farbwert(aktivfa
          rbe,2),farbwert(aktivfarbe,3)
 76 nB0 regler2:
 77 sd2   LINE (posx,py%)-(feldx*6,py%-bahn),aktivfarbe,bf
 78 U0    LINE (feldx*2,py%)-(posx,py%-bahn),zfarbe,bf
 79 FH    px%=feldx*6+bahn: GOSUB curpos
 80 xd    PRINT farbwert (aktivfarbe,i);"  "
 81 Rj    px%=feldx-bahn
 82 gI0 RETURN
 83 wf  curpos:
 84 wa2   POKEW WINDOW(8)+36,px%: POKEW WINDOW(8)+38,py%
 85 jL0 RETURN
 86 QS  END SUB
 87 BE  '*** Ende des Unterprogramms ***
 88 Q6  '*** Demoprogramm ***
 89 s42   Modus=1          :'1 bis 4
 90 1Y    Farbscreens=5    :'1 bis 5
 91 Zg0 screenx= 640 /((Modus AND 1)+1)
 92 L9  screeny= 200 *((Modus+1)\2)
 93 XQ  SCREEN 1,screenx,screeny,Farbscreens,Modus
 94 jG  WINDOW 3,"    F=Farbwahl - E=Ende      ",,2,1
 95 VS  ' ****
 96 oT  GOTO main: 'Diese Zeile ist nach dem ersten Programmlauf zu
        löschen!
 97 XU  ' ****
 98 3f  OPEN "Farbdaten" FOR INPUT AS 1
 99 A9  INPUT#1,farbzahl: DIM farbwert(farbzahl,3)
100 G02   FOR i=0 TO farbzahl
101 wC4     FOR j=1 TO 3
102 PN6       INPUT#1,farbwert(i,j)
103 8P4     NEXT j
104 MS      PALETTE i,farbwert(i,1),farbwert(i,2),farbwert(i,3)
105 80      NEXT i
106 WG0 CLOSE 1
107 sv  main:
108 Ke  WHILE a$<>"e"
109 4X4     LINE (RND(1)*screenx,RND(1)*screeny)-(RND(1)*screenx,RND
            (1)*screeny),RND(1)*WINDOW(6),bf
110 sQ      a$=INKEY$
111 2U      IF a$="f" THEN CALL farbwahl
112 F30 WEND
113 TR  WINDOW CLOSE 3: SCREEN CLOSE 1
114 aV  END
(C) 1988 M&T
```

Listing. Mit »ColorChange« können in Basic-Programmen die Farben beliebig verändert werden. Bitte mit dem Checksummer eingeben.

MESSAGE Computer

Bestellung und Versand:

Telefon (0208) 24047
BTX 020824047
Stöckmannstraße 78
4200 Oberhausen 1

Qualität ist kein Zufall!

Externes Diskettenlaufwerk MAD-V+ für Amiga 500/1000/2000

□ geeignet für 5,25″-Disketten □ Bus durchgeführt □ abschaltbar □ 40/80 Track umschaltbar □ kompatibel zu PC-Karten (Side Car)
880 KByte Speicherkapazität **DM 459,–**

Echtzeituhr MCT-1000

□ Anschluß am Expansionport, Drucker und Joystickport bleiben frei □ akkugepuffert (garantiert 1 Jahr ohne Einschalten des Rechners) □ Schreibschutzschalter gegen versehentliches Verstellen (bei Programmabstürzen) □ inkl. Steuersoftware zum Einbinden in die Startup-Sequenz
□ quarzgenau **DM 98,–**

512 KByte Speichererweiterung für Amiga 500

□ erweitert den Speicher auf 1 MByte □ abschaltbar □ akkugepufferte Uhr optional
auf Anfrage

AHD20-Harddisk für AMIGA

□ volle Amiga-Kompatibilität (Kickstart 1.2) □ vorbereitet für Autoboot-Kickstart und Workbench □ kompatibel zum ST 506-Standard □ kompatibel zu allen Speichererweiterungen am Expansionport (Golem-Box) □ keine Belastung der Amiga-Stromversorgung (eigenes Netzteil ohne Lüfter) □ wahlweise MEM (Standard) oder RLL-Controller (50% höhere Kapazität) □ erhältlich in 20, 33, 40, 66 MByte □ inkl. Steuersoftware und Programmen zur Datensicherung auf Disketten □ Geschwindigkeit gegenüber Floppy bis 10fach (Systemstart 15 Sekunden) □ automatisches Parken des Schreiblesekopfes als Stoß- und Transportsicherung □ Aufteilung in mehrere logische Laufwerke möglich □ inkl. jede Menge Public-Domain-Software ab **DM 1.498,–**

PAL – Genlock – Interface

□ getrennte Regelung von Computer und Videosignal (fade-in/fade-out) □ FBAS und RGB-Ausgang □ für Amiga 500/1000/2000 geeignet □ Amigamonitor ist nutzbar als Kontrollbildschirm □ Titeleinblendung bei Überspielung zwischen 2 Videoquellen **DM 598,–**

Zusatzlaufwerk MAD-II +

□ Panasonic JU363 (Original Amiga-Laufwerk) □ kompatibel zu Kopierprogrammen (Marauder) □ abschaltbar □ inkl. Utility zum Einstellen der Step-Geschwindigkeit für schnellere Suchzeiten und leiseren Betrieb, auch für das interne Laufwerk. **DM 359,–**
mit BUS-Durchführung DM 369,–

MIDIFACE

□ kompatibel zu allen MIDI-Programmen □ für Amiga 500, 1000 und 2000 □ 1 x MIDI IN, 1 x MIDI OUT, 1 x MIDI OUT/THRU schaltbar □ mit Kontrollanzeige für MIDI-IN und OUT, erleichtert Auffinden von Übertragungsfehlern (defekte Kabel) □ stabiles Metallgehäuse mit langem Anschlußkabel (keine nackte Platine an der Rechnerrückseite) **DM 129,–**

Drive-Expansionbox

□ bis zu 3 Diskettenlaufwerke anschließbar □ geeignet für Diskettenlaufwerke ohne Busdurchführung □ abschaltbar □ für 3,5″- und 5,25″-Drives □ Drive 1 + 2 vertauschbar □ keine Kabellängen-Probleme □ stabiles Metallgehäuse, dadurch optimale Abschirmung
DM 79,–

Abschaltung für Speichererweiterung (Amiga 500) **DM 29,90**

Centronics-Druckerkabel für Amiga 500/1000/2000 **DM 39,–**

Filialen/Ladenverkauf/Stützpunkte:

2000 Hamburg 62,
Langenhorner Chaussee 670b, Tel. (040) 5276404, K. Engler, W. Sonnemann

4200 Oberhausen,
Stöckmannstr. 78, Tel. (0208) 24047, Thomas Martin

8541 Rohr-Regelsbach,
Tel. (09122) 82563, Robert Esser

HÄNDLERANFRAGEN WILLKOMMEN!

Österreich: *SUETRAK HANDELSGES. M.B.H. · Mitterauen 31 · A-3003 Gablitz · Tel. 02231/2170*
Schweden: *PLAMI PRODUKTER · Box 104 · S-27400 Skurup · Tel. 0411-32260*
Dänemark: *ABSALON DATA · Vangedevej 216A · DK-2860 Söborg · Tel. 01671193*
Versand per Nachnahme · Ausland nur Vorauskasse + DM 10,–

MK – I

Kickstartumschaltung für Amiga 500/2000.

Einbau ohne Löten □ voll steckbar □ inkl. Software, die brennfertige Files von Ihrer Kickstartversion erstellt.

Leerplatine **DM 39,–**
Fertiggerät **DM 98,–**

Eprombrennservice für Kickstartumschaltung MK-I **DM 39,–**

AMIGA COMPUTER-MARKT

Wollen Sie einen gebrauchten Computer verkaufen oder erwerben? Suchen Sie Zubehör? Haben Sie Software anzubieten oder suchen Sie Programme oder Verbindungen? Der COMPUTER-MARKT von »Amiga« bietet allen Computerfans die Gelegenheit, für nur 5,— DM eine private Kleinanzeige mit bis zu 4 Zeilen Text in der Rubrik Ihrer Wahl aufzugeben. Und so kommt Ihre private Kleinanzeige in den COMPUTER-MARKT der **Juni-Ausgabe** (erscheint am 25. Mai 88): Schicken Sie Ihren Anzeigentext bis zum 19. April 88 (Eingangsdatum beim Verlag) an »Amiga«. Später eingehende Aufträge werden in der **Juli-Ausgabe** (erscheint am 29. Juni 88) veröffentlicht.

Am besten verwenden Sie dazu die vorbereitete Auftragskarte am Anfang des Heftes. Bitte beachten Sie: Ihr Anzeigentext darf maximal 4 Zeilen mit je 40 Buchstaben betragen. Überweisen Sie den Anzeigenpreis von DM 5,— auf das Postscheckkonto Nr. 14199-803 beim Postscheckamt mit dem Vermerk »Markt & Technik, Amiga« oder schicken Sie uns DM 5,— als Scheck oder in Bargeld. Der Verlag behält sich die Veröffentlichung längerer Texte vor. Kleinanzeigen, die entsprechend gekennzeichnet sind, oder deren Text auf eine gewerbliche Tätigkeit schließen läßt, werden in der Rubrik »Gewerbliche Kleinanzeigen« zum Preis von DM 12,— je Zeile Text veröffentlicht.

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

Suche: Software

Suche Sidecar, zahle je nach Software bis 700 DM. Tel. 07071-73238 ab 17 Uhr

Gesucht: Tauschpartner (Amiga). Eigenschaften: schnell, zuverlässig. Besoldung: (gute) Software. Wenden an: 05103/7507

Suche deutsche Anleitung für den Aztec (CV3.4a (od. spät. Version), sowie Bücher über C. Suche orig. Soft. Tausche PD-Soft. Bitte melden bei Sascha, Tel. 05130/5161

Suche A1000 Software, Anwender-Spiele, zahle f. Originale je nach Alter 50—70% v. NP. Suche auch Festplatte. Preis: VB. G. Pekers, Langenburger 7, 7118 Künzelsau

Prg., mit dem ich Geschäftsgrafiken zeichnen kann (Säulen, Torten usw.), auch jede Anwendung für Landkarten gesucht. Bin nur Mo.-Do. erreichbar, 07154/4508, Rainer

Suche div. Programme u. Anleitungen für Amiga 500. Suche Druckertreiber für Seikosha GP-700 A. Jürgen Hamacher, Sandstr. 15, 5010 Bergheim 3

Wer schenkt armen Schüler Pascal oder Modula II Compiler? Wenn möglich mit Anleitung. Danke! Ralf Wirth, Meßbergstr. 9, 5757 Wiche-de/Ruhr

Amiga. Suche Software, z.B. Samanta Fox, Printmaster+, Astro, Programm des Lebens, Fontmaster, Demo Creator, TV Text usw. Listen an S. Reichel, Senftenberger Ring 42d, 1000 Berlin 26

Amiga 500
Suche Software für Vermessungstechniker und gute Matheprogramme, Klaus Risken, Bruchstraße 7, 4770 Soest-Enkesen, Tel. 02928/291

Tausche für A500 Software, Anleitungen, suche zuverlässigen Tauschpartner (auch Anfänger). Schreibt: Möhwald Fritz, 8481 Altenstadt/Wn, Kantstr. 6, Hi Stefan.

Suche preisg. neuste Version De Luxe Video evtl. im Tausch gegen orig. DPaint II mit Handbuch neueste Version. H. Niemeyer, Tel. 0511/691845

Amiga-Einsteiger sucht zuverlässigen Softwaretauschpartner. Wer hat den Amiga-Sort aus Amiga 1/88 abgetippt? Dirk Dreyer, Igelweg 19, 6090 Rüsselsheim

Suche: Scenario Discs und Flugkarten für den Flight Simulator II! 05451/16210. Meldet euch! Ruft an! 05451/16210

Hello Amiga-User-Family! Suche Software und Anleitungen aller Art, besonders Buchführungsprogramm u. Amiga Art Machine-Produkte. J. Steinmaßl, Birkenweg 9, 8221 Taching

Suche Amiga-Programmierer u. Programme. Keine Raubkopien. Nur selbstgeschriebene Programme! Ruft doch einfach mal an! 089/1783101 möglichst gegen 19 Uhr!

Habe und suche neueste Soft. Call: 02151/544511 (Ingo)

Suche Software für Amiga 500. Einnahme-/Überschußrechnung, Lagerhaltung, Faktura, Adressenverwaltung. Tel. 02104/52561

Suche Software und zuverlässige Tauschpartner. Listen an Elmar Hoppe, Begonienstr. 14, 4600 Dortmund 41. Tel. 0231/400351

Amiga 2000-Einsteiger sucht Software und Kontakte zu anderen. Erwin Gloe, Pestalozzistr. 3, 7750 Konstanz

Suche Software aller Art für A500. Liste an: P. Zingelmann, 2410 Mölln, A.d. Hermannsquelle 1c oder Tel. 04542/7808 ab 18 Uhr. PS. Möglichst m. Anleitung für Anw.-Prg.

Suche günstig Software für Amiga 500. Listen bitte an Thorsten Bopp, Goldbachstr. 54, 6991 Igersheim

Ich suche neue Software zum Tauschen! Amiga, C64, IBM, 520 ST, 100%ige Antwort. Bei: Bobby Lauterwein, Nachtigallen 12, 4400 Münster. Call: 0251/614571 sofort

OK, the World Famous Sudl is searching for new Contacts all over the World! Write to: Tom Seidel, Herrenstr. 24, 8360 Deggendorf or call: 0991/23693! Answer 100%!

Suche günstig Software f. Amiga 500 (Spiele, Adressverwaltung, Video, Lager, Grafik). Angebote an: Dieter Pischke, Doheerm.-Höhe 40a, 4830 Gütersloh 1

Für A500 zu kaufen gesucht: Gute Spiele und Anwendersoftware. Tel. ab 18 Uhr 07141/44633, R. Werner, Engelbergstr. 8, 7140 Ludwigsburg

Suche zuverlässige Tauschpartner für Amiga 500 + eventuell 512-K-RAM-Erweiterung (günstig). Listen + Angebote an: Manfred Mielke, 5010 Bergheim, Otto-Hahn-Str. 6

Suche Tauschpartner für Amiga! Dirk Barth, Heppenheimer Str. 35, 6149 Fürth, 06253/21267, Christoph Arnold, Alfred-Delp-Str. 1, 6149 Fürth/Odw. 06253/5755

Suche Tauschpartner für Amiga-Software, Tel. 0451/70210. Nach Maik fragen (ab 17 Uhr)

Suche Programme aller Art
Vor allem De Luxe Musik, Print II, Video, Print. Zahle bis 40 DM oder wenn möglich umsonst. Schr. an Ralf Schmidt, Störr 11, 2213 Wilster

Ich suche Bücher über Amiga 500 und Programmierkurse in Basic und C. Welcher Amiga-Freund ist mir behilflich! Tel. 069/374521. Alain Branchereau, Farbenstr. 98a, 6230 Frankfurt 80

Suche billige Software für Amiga 500. Liste bitte an T. Varadi, Schießstattstr. 16a, 8200 Rosenheim

Computergirl sucht für Amiga 500 Software (De Luxe Paint II, De Luxe Musik, Flugsimulator II) sowie gute Spiele und einen Digitizer mit Software. Andrea Alt, Buchenweg 6a, 8521 Großenseebach

Suche Kontakte zu netten Amiga-Anwendern zwecks Softwaretausch. Bevorzugt Raum Stuttgart u. Umgebung. Eddy wartet auf Anruf ab 18 Uhr. 07142/42907

Bau-Software gesucht. Angebots-Rechnung-Ausschreibung, Haushaltsverwaltung, Steuer auf Amiga 500. Bernhard Maier, Sommerberg 37, 7801 Buchenbach

Suche Software für Wohnungseigentums-Verwaltung (Amiga 500). Angebote an: H. Mölken, Bredeneger Str. 26, 4300 Essen 1

Suche Spiele (Original) Superkit, Bad Cat, Faery Tale, Auto Dael, Go Amiga Datei. Zahle bis 40 Mark, wenn möglich umsonst. Schr. an Ralf Schmidt, Stöhr 11, 2213 Wilster

We are looking for new Contacts. Call: 0991/8414

Suche Tauschpartner für Software A500. Tel. 05254/10202. Bitte in der Zeit von 15 bis 21 Uhr. Danke

Waasss! Ihr habe noch nicht die neuste Software? Dann mal schleunigst in München anrufen und Contacts knüpfen! Call: 089/3006915 (Ask for Felix!!)

Anfänger sucht Software für Amiga 500. Schreibt an: Andreas Votres, Dorumer Weg 44, 2850 Bremerhaven oder ruft am besten am Wochenende unter 0471/51284 an

Erfahrungs- und Softtausch
mit Einsteigern und »alten Hasen« ges. U. Anussek, Elsa-Brandström-Str. 7, 7942 Zwiefalten. Freue mich auf Post

Suche Lattice C und Aztec C (jeweils neueste Version und mit Anleitung)! Angebote an: H. Bielenstein, Cloppenburger Str. 300, F4, 2900 Oldenburg

Tausche Public Domain Software! Ich habe 18 Amicus- und 3 Faug-Disks.! Write or call: 02204/72347 (14—16 Uhr)! Michael Will, Am Fischweiher 2, 5063 Overath 8

Suche PD für A500. Wer hat Börsenprogramm für A500? Angebote an: Hans-Dieter Beck, Böhmerlandstr. 5, 8752 Mömbris

MS-DOS Amiga
Tausche MS-DOS und Amiga Software, auch PD. Tel. (Germany) 02156/3985 (Bernhard)

The Skull ist searching for Amiga-Contact. Please call Germany 05621/73417

Suche und tausche Software für Amiga 1000 und 500. Listen an C. Ükermann, Heinrichstr. 1, 3504 Kaufungen 1, Tel. 05605/12897 (ab 15 Uhr)

Suche Tauschpartner für Amiga 500. Tel. 07051/7319

Amiga 1000-Besitzer sucht Amiga-Club zwecks Erfahrungsaustausch usw. Bitte Zuschriften an A. von Sturm, Groß-Flottbeker-Str. 30, 2000 Hamburg 52

Suche Software aller Art. Hauptsächlich Spiele. Liste mit Preise an Peter Stock, 4290 Bocholt, Thüringer Str. 33

Fortgeschrittener sucht Software! Habe 61 Disks. Bin an Tauschpartnern interessiert! Tel. 02134/34674, nur vor 18 Uhr!

Suche Tauschpartner für Amiga-Soft Raum HD/MA, auch Anfänger. Schwerpunkt: Musik, MIDI, gute Spiele. Holger: 06221/471626

Suche Tauschpartner für A500. Artur Urbat, Hahnbusch 14, 2300 Kiel 14. 100% Antwort!

Suche Kontakt zu anderen A500-Usern. 100% Antwort. M. Urbat, Hahnbusch 14, 2300 Kiel 14

Suche Tauschpartner für Amiga. Kaufe auch Copyrights von guten Spielen. Suche auch RAM-Erweiterung, usw. Bitte melden bei: 05103/7507

Suche Software für A500. Searching worldwide connections. Torsten Schmidt, Danziger Allee 81d, 6203 Hochheim

Suche Sidecar. Zahle je nach Software bis 700 DM. Tel. 07071/73238 ab 17 Uhr

Suche und tausche neueste Software. Schreibt und schickt Eure Listen an: Rüdiger Kurz, Faunastr. 27, 4000 Düsseldorf. I'm working with an Amiga 5000, Shorty

Suche Druckertreiber für CBM-MPS 2000 C oder für NEC CP6 sowie Original Grabbit und Original Videoscape 3D (dt.): R. Vogel, Tel. 0551/66945

Suche Tauschpartner(in) für Amiga PD-Soft! Schreibt an (mit Liste): Sascha v.d. Hellen, Berggrund 8, 3002 Wedemark1. Oder ruft an: T. 05130/5161, am besten zwischen 19 bis 20 h.

Suche preisgünstige Amiga-Software aller Art. Ruft an ab 17 Uhr 02534/2447 oder schreibt an Boris Bratz, Tischlerweg 10, 4400 Münster

Suche Tauschpartner für Amiga-Software. Tel. 02641/3305 (Markus) ab 14 Uhr

Zahle gut: Suche deutsche Anleitung für Aztec-C-Compiler. Tel. 0231/483382 (Ingo)

Suche gutes Skat-Prg. für Amiga 2000. Wolfgang Neuhoff, Jagowstr. 8, 1000 Berlin 21

Suche günstig A500-Spiele und Amiga-Glossar, Ausgaben 9/87—12/87 gegen Unkostenerstattung, schreibt an: L. Breitzke, Wallbergstr. 2, 8017 Ebersberg

Anfänger sucht Software für Amiga 500. Listen an: Rainer Tobsch, Postfach 241, 8221 Inzell

Suche Tauschpartner für A500. Wer hat DPrint + DVideo 1.2? Schreibt an: Daniel Schlachter, Ringstr. 17, 8947 Mkt. Rettenbach

Suche Software aller Art. Tausch oder Kauf. Suche C oder Modula. Michael Kreutzmann, Elbestr. 22, 4550 Bramsche 1

Suche dringend jegliche (alle) Anleitungen zu Software. M. Knöthig, Am Sandberg 28, 5270 Gummersbach

Du hast neueste Amiga-Software? Ich auch! Ruf an: Jörg 09856/1400

Achtung:

Wir machen unsere Inserenten darauf aufmerksam, daß das **Angebot**, der **Verkauf** oder die **Verbreitung** von urheberrechtlich geschützter Software nur für Originalprogramme erlaubt ist.

Das Herstellen, Anbieten, Verkaufen und Verbreiten von **»Raubkopien«** verstößt gegen das Urheberrechtsgesetz und kann straf- und zivilrechtlich verfolgt werden. Bei Verstößen muß mit Anwalts- und Gerichtskosten von über DM 1 000,— gerechnet werden.

Originalprogramme sind am Copyright-Hinweis und am Originalaufkleber des Datenträgers (Diskette oder Kassette) zu erkennen und normalerweise originalverpackt. Mit dem Kauf von Raubkopien erwirbt der Käufer auch kein Nutzungsrecht und geht das Risiko einer jederzeitigen Beschlagnahmung ein.

Wir bitten unsere Leser in deren eigenem Interesse, Raubkopien von Original-Software weder anzubieten, zu verkaufen noch zu verbreiten. Erziehungsberechtigte haften für ihre Kinder.

Der Verlag wird in Zukunft keine Anzeigen mehr veröffentlichen, die darauf schließen lassen, daß Raubkopien angeboten werden.

Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen Private Kleinanzeigen

Suche f. Amiga 500 (ohne Erw.) gebr. Hardware u. Original-Software aus allen Bereichen, mögl. mit Anleitungen. Tel. 0234/799757 (öfter versuchen!)

Suche Software aller Art für den Amiga 500, möglichst mit Anleitung. Bitte keine Tel.-Anrufe! Anschrift: Alexander Richter, Kantstr. 2, 6094 Bischofsheim

Suche zuverlässige Tauschpartner(innen). 100% Antwort. Listen an: Daniel Eiba, V.-Langenmantel-Str. 8b, 8902 Westheim. All over the Globe!

Hallo Amiga-Freaks. Ich suche den Flight-Simulator für Amiga mit deutscher Anleitung. Meldet Euch bitte bei 06131/225185, Thomas verlangen. Vielen Dank!

Computergirl sucht Software für Amiga 500. De Luxe Paint II, De Luxe Musik, Flugsimulator II sowie gute Spiele und einen Digitizer mit Software zu akzeptablen Preisen. Andrea Alt, Buchenweg 6a, 8521 Großenseebach

Computerclub International
Deutschlands führender Club für alle AMIGA-User. Wir bieten: Clubzeitschrift, PD-Software, Beratung, Einkaufsvorteile und vieles mehr!! Beitrag nur DM 3,33 pro Monat!! Info gegen DM 0,80 Rückporto bei: CCI, Dorstener Str. 31, 4350 Recklinghausen. Tel.: 02361/15943.

Ausland

Suche Tauschpartner in aller Welt. Liste an: Hofstätter Hans, Blumengasse 7/19, A-1170 Wien

Habe/Suche/Verkaufe Soft.-Liste an T. Martinelli, Buchwiesen 63, 8052 Zürich, Tel. 01/3024185. Searching for new Contacts in the whole World. Please call me after 17 h

Austria-Amiga. Suche und tausche Software aller Art. Listen an Schwaighofer Gerhard, A-3383 Hürm Nr. 16. 100% Antwort

Amiga Freaks! Suche Top-Tauschpartner für Top-Software. Daniel Bucher, Hirtenhofstr. 25, 6005 Luzern, Schweiz

Suche A2000-SW Raum Süddeutschland. R. Sulzer, Gstaltenrainweg 74, CH-4125 Riehen/Schweiz

Swedish Amiga-Owners are looking for Contacts abroad. Contact: Jonas Sanomon, Box 16, 67010 Töcksfors, Sweden/Ulf Berg, Box 5153, 66400 Grums, Sweden

Suche Tauschpartner für Amiga. Anfragen bitte schriftlich (mit Liste/Prg.) an: Biwer Patrick (Marcel) Inst. St. Willibrord, L-6401 Echternach/Lux.

Amiga Schweiz
Suche Hilfe, suche Spieltauschpartner aus aller Welt. Listen an: Sascha Imboden, Postfach 30, CH-6373 Ennetbürgen, 041/642268

Verkaufe/tausche die wirklich allerneueste Software, auch das alleräIteste vorh. (reichlich vorhanden!). Bei Interesse: J.-J. Wagner, Schönbühl, CH-3807 Iseltwald (CH)

Biete an: Software

Verkaufe Originale: Programm des Lebens (Astrologie), Deluxe Video V1.1 beide mit Anleitung, à 100 DM. Tel. 0711/875957. Barthelmeß N., Markgröninger Str. 42, 7000 Stuttgart 40

Tauschen macht Spaß! Komm und mach mit. Software für Amiga + C64/128. Hotline. 02843/3576. Jetzt anrufen!!

Tauschpartner für Amiga gesucht (auch Anfänger). Call 02323/8590 (Robert)

Verkaufe original Spaceranger und Phalanx 2, jeweils 20 DM. Bin an Kontakten in Ffm. oder Marburg interessiert. Tel. 069/595040 (Manfred)

PD Software: Info bei: D. Kommelter, vom-Bruck-Platz 45, 4150 Krefeld

Biete und tausche aktuelle Software. Frank Olepp, Auf Brohl 7, 5476 Niederzissen. Tel. 02636/6217

Alle tauschen! Du auch? Software für Amiga + C64/128 Hotline: 02843/3576, jetzt anrufen!

Tausche Top Amiga-Soft, kaufe Originale/Anleitungen. Michael Dettmer, Bromerstr. 24, 3177 Grussendorf

Tauschpartner für Amiga Software gesucht. 07832/3529

Hey Amiga Anfänger! Sucht Ihr preiswerte, aktuelle Software? Dann ruft an: 07832/3529

Suche Tauschpartner, tausche auch Amiga-Soft gegen ST-Soft! Verkaufe deutsche Videoscape-Anleitung. 05657/1650

PD-Software wie Kick 1.1-Emulator, Diashows etc. + 68020 + 68881 Piggy-Board 12.5 MHz. Info bei: T. Wirz, Höhenweg 98, 5300 Bonn 1 (alles einzeln!)

Habe und suche neueste Soft!! Call: 02151/544511 (Ingo)

Suche Tauschpartner für Amiga + C64 Software! Listen an: Oliver Kruschat, Gartenstr. 4, 5828 Ennepetal

Amiga-Originale! Quiwi 30 DM, Roter Oktober, alles deutsch 50 DM, Leviathan 40 DM, Miss. Elevator 35 DM, Leader-Board 1.1 35 DM, Data-Becker Einsteiger 25 DM. 0211/721159, Joe

Amiga-Art-Machine! Disc I+II Animation & Objekte, Bilder à 13 DM, Disney-Demos, Smurf/Head/Cat à 8 DM, Dpaint II, Anwender/deutsch 25 DM. Tel. 0211/721159 ab 14 Uhr, Joe

Amiga, ST, C64, IBM. Ich tausche und verkaufe Soft! Schickt zu: Bobby Lauterwein, Nachtigallen 12, 4400 Münster. Listen + Disks schicken! Call: 0251/614571

Gebe günstig ab: PRG-Sammlung für Atari ST und Amiga, je mehrere Disks. Info 80 Pf. Suche Kontakt zu IBM/AT-Usern. Hans Becher, Tiegenhöferstr. 10, 5603 Wülfrath

MPS802 druckt Texte von Amiga über C64. Info gegen frankierten Rückumschlag zu senden an: Oliver Bürk, Lupinenweg 4, 7263 Bad Liebenzell

Logistik, deutsch, Tabellenkalkulation, Datenbank, Zeitplan, Grafiken, neu, 25% unter Neupreis. TV-Modulator 35 DM. 05033/5775

Amiga 1000, 512 KB + Farbmonitor 1081, deutsche Tastatur + Bücher + Software 1800 DM. Sidecar m. 20 MB Filecard, Uhrenkarte, V.24, Software, 1900 DM. Tel. 0228/624256 ab 17.30

Orig. m. Anl.: Marble Madness 35 DM, Emerald Mine 19 DM, Terrorpods 40 DM, BardsTale I 43 DM, Defender o.t.Crown 40 DM, Balance of Power 40 DM, * alles in Superzustand * — Epyx(!)-Joystick XJ500 (Microschalter) 28 DM. Zeitschriften je 2 DM. 9 BASF PD-Disks 2DD 35 DM. Tel. ab 15.30 unter 06622/2988, Andreas

Suche neue Kontakte und Tauschpartner für Amiga-Software. Ich habe keine finanziellen Interessen. Ruft an: 02641/36435

Verkaufe Amiga-Paint, IFF-kompatibles Malprogramm mit deutscher Anleitung und 14 IFF-Bildern für 30 DM. Tel. 0281/70290

Suche Programme aller Art. Keine Anfänger. Nur Tausch kein Kauf. 08663/1871

Suche zuverlässigen Tauschpartner. Genügend Software vorhanden. Pointer Anton jun., Gstatt 1, 8222 Ruhpolding, 08663/1871

Verk. Superbase (deutsch) wenig benutzt, ein Monat alt, wieder original eingepackt (ca. 200 DM). Verk. auch Fred Disk Nr. 99 + Tora. 3. Michael Krämer, Burgwall 52, 4400 Münster

Tausche Amiga Public Domain Soft. Ich habe 18 Amicus u. 3 Faug. Write or call 02204/72347 (14 bis 16 Uhr). Please fast! M. Will, Am Fischweiher 2, 5063 Overath 8

The Farmers haben und suchen die neuste Top-Software für den Amiga (auch Anfänger) Contact: 05621/71723 (Thomas), 05621/5589 (Steffen)

Verk. ungebr. Originale von Seven Cities of Gold (Kickst. 1.1) zu 25 DM u. Cruncher Factory zu 10 DM. Suche Listen für PD. Schreibt an: A. Bartl, Bernsteinstr. 74, 7000 Stuttgart 75

Suche Tauschpartner für Amiga-Soft (auch Anfänger). Besonderes Interesse: Musikprogr. 06221/471626, H. Nißle, Trübnerstr. 14, 6900 Heidelberg

Wegen Zeitmangel verkaufe ich komplette Programmsammlung. Info gegen Rückporto. K. Weger, Kantstr. 25, 4005 Meerbusch 1

Biete, suche und tausche PD- und sonstige Software aller Art. Tel. 0711/544607 ab 18 Uhr

Call for hot Stuff: 04101/43254 Oliver

Verkaufe original Amiga-Software Calligrapher Fonteditor 140 DM, Deluxe Paint II 140 DM. 040/6526152 ab 17 Uhr

Biete folgende Originale zum Tausch an: Strip Poker, Emerald Mine, Rocket Attack, Pac Boy, Shooting Star. Suche u.a. Infocoms. Tel. 05204/8196

Original Amiga Software Calligrapher Fonteditor 140 DM, De Luxe Paint II 140 DM. 040/6526152, D. Herzog, Wandsbeker Marktstr. 42, 2000 Hamburg 70

PD Soft und andere Info von J. Klein, Herrenstrunden 46, 5060 Berg.-Gladbach 2. Gruß an alle Amiga User und die es werden wollen!

Verkaufe orig. Word Perfect 4 Disketten + Handbuch für 500 DM unbenutzt oder tausche gegen orig. Acquisition Datenbank. Tel. 09082/2199 ab 17 Uhr

Waaasssss??? Ihr habt noch nicht die neuste Software??? Dann mal schleunigst in München anrufen und Contacts knüpfen! Call: 089/3006915 (ask for Felix!)

Original-Software 50 % unter akt. Preis, DPaint2, DigiPaint, Prism, PageSetter, Grafiktablett ProwDraw, A4, 480 DM. Tel. 0221/217384

Hey Freaks!
For the latest Amiga Stuff call fast: 04101/43254 Oliver

Do you want Swap the hotest Stuff on Amiga?? Then Call: 04101/43254 Oliver

Top-Software
Biete und tausche. Detlef Paulus, Brunsbütteler Damm 64, 1000 Berlin 20

Call 04101/47730, for the hotest stuff in town

Du hast neue Software? Ich auch! Ruf an: 09856/1400

PD-Softw. z.B. Fish, Kickstart 1.1-Emulator Diashows, Grafik-Disks. T. Wirz, Höhenweg 98, 5300 Bonn 1. Außerdem Turbokarte 68020 + 68881 12, 5 MHz für A1000 + 2000 + 500, VB 1495 DM

Verkaufe/Tausche neuste Top-Amiga-Soft! Schickt eure neusten Games an: Daniel Wolf, Moltkestr. 42, 4902 Bad Salzuflen, Antw. 100%.

Contact: 04193/3456. For hot Soft!

Verkaufe Originale: Superbase Amiga für 175 DM, Lattice C-Comp. 4.0 für 340 DM, für MS-DOS BCI-Pascal für 175 DM (NP 249). Jürgen Hilbich, Hoscheiter Str. 12, 5107 Simmerath, 02473/6218

Suchen/Haben/Tauschen/Verkaufen. Soft-Hardware, Kontakte für alle Computer + Video z.B. Amiga, Atari etc. Info: für 80 Pf. bei: CMP-Club, Wiechertstr. 34, 4030 Ratingen 1

Immer neueste Software. Tel. 0291/3216

Verkaufe Amiga-Karate für 500er 30 DM. Kaufe preiswerte Programme m. Anleitung, auch Public Domain. Alexander Heukäufer, Kirchstr. 2, 5240 Alsdorf

Hallo Public Domain-Freaks. Ich suche nette leute zum Tauschen, nur PD. Auch Verkauf. Schreibt an: GPP, Linnenbecker Weg 7, 4973 Vlotho. Keine Anrufe! Ciao!

Verkaufe neue Originale aus GB und USA, wegen Hobbyaufgabe. Habe aber alles was rauskommt direkt in GB und USA bestellt. Wolfgang Jesberger, Schelmenw. 4, 7916 Straß

Achtung Top Amiga Software. Neue Software die noch keiner hat! Tausche nur Top-Software! Meldet euch bei Schwaiger Alexander, Am Zellerberg 18, 8222 Ruhpolding

Original-Software gegen Höchstangebot: Marauder II, Chessmaster, Starglider, OGRE, UBM-Text. Rainer Baumann, Schillerweg 28, 7186 Blaufelden

Originaldisk: Sculpt-3D für 150 DM. M. Groß 08131/82525

Wichtige Hinweise für alle Kleinanzeigeninserenten:

★ Kleinanzeigenaufträge **ohne Absenderangabe auf der Rückseite der Karte**

sowie

Anzeigentexte unter **Postlagernummer** können leider **nicht** veröffentlicht werden.

★ Zur Bezahlung von Kleinanzeigen können ab sofort **keine Fremdwährungen** mehr angenommen werden.

★ Bitte achten Sie auch darauf, daß Ihre Auftragskarten **immer vollständig** ausgefüllt sind (z.B. Unterschrift)



Private Kleinanzeigen

Amiga-User aufgepaßt!
Suche/Tausche Amiga-Software, egal ob Spiele oder Anwendersoftware. Listen an Wirth Karl, Kirch-Str. 55, 8533 Scheinfeld

Original-Software Zing! Keys 75 DM. Amiga 500 Buch von Markt & Technik 35 DM. Das können Amiga 500 & 2000 Buch 20 DM. Tel. 05361/66159 ab 19 Uhr

Word Perfect V4.1 Orig. supergünstig zu verkaufen, da für mich zu umfangreich. Das Textverarbeitungsprogramm, 4 Disk, NP 790 DM nur 580 DM. Tel. 02351/24502

Amiga-Software-Tauschpartner gesucht. Wer besitzt Intro-&-Demomaker? Listen an S. Reichel, Senftenberger Ring 42 d, 1000 Berlin 26

Maxiplan plus 1.8 280 DM, Phantasie III 55 DM; Emerald Mime 15 DM, Feud 15 DM, Ultima IV 80 DM; Moebius 60 DM, Design Text 300 DM, ggf. Tausch, ab 19 Uhr 0209/200520

Kick start
1987 komplett (6 Ausgaben, wie neu!) zu verkaufen. Angebote (nur schriftlich) an Olaf Winkler, Uhlandstr. 7, 7889 Grenzach-W.

Verschenke für Amiga und PC-Kompatible Software gegen UKB von 4 DM/Disk. Liste gegen 3 DM. O. Saenftl, Roßmarkt 30, 7300 Esslingen. Tel. 0711/357908

The Farmers haben und suchen neuste Top-Software für den Amiga. Contact: 05621/71723 (Thomas), 05621/5589 (Steffen)

Verk. Superbase (deutsch), ein Monat alt, wieder original eingepackt (ca. 200 DM). Verk. Fred Fish Nr. 99 + Tornados Nr. 3 für 8 DM. Krämer, Burgwall 52, 4400 Münster

Verkaufe Amiga-Paint, IFF-kompatibles Malprogramm mit deutscher Anleitung und 14 IFF-Bildern für 30 DM. Tel. 0281/70290

Ausland

Biete günstig neue Soft. Call Darth Vader nur Fr. 19 bis So. 19.30 Uhr durchgehend online. Tel. Austria 0043/022312850. Auch PD oder Tausch. Bis bald! Suche Kontakte!

Anfänger — Profis! Software aller Art, superpreisgünstig abzugeben (Tausch?!?) Erfahrungsaustausch! Antworte zu 100%. M. Sepp, Roseggerstr.4, A-6020 Innsbruck

Amiga-Soft-Schweiz
Wir bieten Soft zu Spottpreisen! Call: 01/9204942 or 01/9201951

Amiga Schweiz! Tauschpartner gesucht! Newest Stuff to swap! Write to: Peter Erni, Riedweg 16, 4800 Zofingen, Schweiz! Enjoy Call: 062/514266. Enjoy it!

Amiga! Atari ST! IBM!
Newest Software write to: Salvatore Innaimi, Via Pentone 5, 00173-Roma-Italia or call: 0039/67971250

Habe/Suche/Verkaufe Soft.-Liste an T. Martinelli, Buchwiesen 63, 8052 Zürich, Tel. 01/3024185, searching for new contacts in the whole world. Please call me after 17 Uhr

Hot Amiga 500 Stuff!
Send List (no Beginners, only great GUYs). Janseggers Marc, Stationsstraat 22, B-2640 Niel, Belgien

Rainer Thomas, Werftenstr. 55, A-9210 Pörtschach. Tel. 04272/263613. Newest Stuff!

CH Amiga
Tausche/kaufe/verkaufe alte & neuste Soft. Listen verlangen bei: Tartarus, Postfach 348, 8038 Zürich, Tel. 01/4815694

Verkaufe/tausche die wirklich allerneuste Software, auch das alleräIteste vorh. (reichlich vorhanden!). Bei Interesse: J.-J. Wagner, Schönbühl, CH 3807 Iseltwald

Habe immer die neuste Software! Newest = Latest! Auch ätere(ste) vorhanden. Schreibe bei Interesse an: J.-J. Wagner, Schönbüchl, CH-3807 Iseltwald, bis bald

Du suchst Software für Amiga? Software aller Art preisgünstig abzugeben (Tausch?) Antwort 100%. Mario Sepp, Roseggerstr. 4, A-6020 Innsbruck

Habe immer topaktuelle Amiga-Software, sehr günstig. Call: 0041/062522948, CH

Habe immer topaktuelle Amiga-Soft; sehr günstig. Call: 0041/062/522948, CH

Biete günstig neuste Soft. Call Darth Vader nur Fr. 19 bis So. 19.30 Uhr durchgehend online. Tel. Austria = 0043/022312850. Auch Tausch.

To Swap hot News on Amiga. Contact me: Eric (TNT) 13, Rue Gounod, 62160 Bully-Les-Mines/France. All Countries are welcomed

Suche: Hardware

Suche Druckertreiber für Commodore-Drucker CBM-MPS 2000 C. R. Vogel, Tel. 0551/66945. Suche auch original Videoscape 3D (dt) und Original Grabbit

Suche Fortran Compiler für Amiga oder Atari ST. Tel. 02452/89378

Wer will schnell defektes Comp.-Zubehör loswerden? Auch gegen Bezahlung. Tel. 04194/511, melden bei R-occo

Suche Amiga 500 + Monitor + Zusatzfloppy + Software für ungefähr 530 DM (2 Monate Garantie). Tel. 05172/7004 nach 16 Uhr, bitte um Verständnis. Danke!

Suche Amiga 500 mit oder ohne Gehäuseschaden aber 100% o.k. (wenn es geht mit Software). Angebote an Klaus Köpernick, Hermann-Berndes-Str. 15, 6507 Ingelheim

Defekter oder preiswerter A500 dringend gesucht! Tel. ab 18 Uhr oder Sa./So. 02254/81210

Suche Amiga 500 o. 1000 (PAL) in einwandfreiem Zustand. Zahle gut, aber keine Phantasiepreise. Angebote mit genauen Preisvorstellungen an: Jürgen Hilbich, Hoscheiter Str. 12, 5107 Simmerath, Tel. 02473/6218

Kaufe sofort: C64/1541/1571, Amiga 500—2000/1081-84, PC10 II/III etc. Manfred Fuchs jr. 0202/469243

Suche Plotter, Drucker; gewerbliche und private Angebote an: P. Sprus, 5303 Bornheim 3, Beethovenstr. 40, Tel. 02227/6816

Kaufe Amiga
Suche Amiga 2000 mit Monitor 1081/1084! Wenn möglich unter 2000 DM, melden bei Christian Hof, Tel. 0941/22912, 8400 Regensburg

Wer verkauft an Fan in der DDR sehr preiswert Comp.Commodore oder Atari, evtl. m. Drucker, Laufwerk! Softwaretausch gewünscht. Kontakt über 05326/3970

Commodore 64 ohne Zubehör (!!!) zu kaufen gesucht (Preis VB). Anrufen täglich von 17 bis 21 Uhr bei 06451/21958 (Michael) PS: Nehme keinen Schrott!!!

Wer schenkt armen Schüler Cray X-MP mit Zubehör + Peripherie + guten Spielen? Kann auch defekt sein! Rainer Koch, O. Hantschel, Birkenweg 2, 8951 Rieden

Suche VD3 Amiga-Video-Digitizer. Angebote an: Michael Hager, Dr.-Schier-Str. 13, 6670 St. Ingbert

Suche VD3 Amiga-Video-Digitizer. Angebote an: Michael Hager, Dr.-Schier-Str. 13, 6670 St. Ingbert, Tel. 06894/8300 (ab 17 Uhr).

Suche f. Amiga 500 (ohne Erw.) gebr. Hardware u. Original-Software aus allen Bereichen, mögl. mit Anleitungen. Tel. 0234/799757 (öfter versuchen!)

Wer hilft mir? Tausche ca. 2 Jahre alten C64/1541/100 Disketten gegen irgendeinen Amiga. Zahle bis zu 100 DM dazu. Help! Amiga! Tino Wienand, 06024/9981

Suche Sounddigitizer für Amiga 2000. Möglichst mit Anleitung. Schreibt an Joachim Oesterle, Kastellstr. 20, 7063 Welzheim. Keine Anrufe bitte!

Suche Fortran Compiler für Amiga 500 oder Atari ST. Verkaufe C64 Görlitz Interface sowie C64 Lit. Tel. 02452/89378

Ausland

Suche Amiga 500, Monitor 1081, (Farb)Drucker und 2. Laufwerk! Zahle gut! Angebote an: Frodo Bosch, Graan voor Visch 19354, 2132 WE, Hoofddorp, Holland, 02503/32891

Biete an: Hardware

Sidecar
neu, original verpackt. MS-DOS 3.2 u. GW Basic. Preis 800 DM, Tel. 0208/842177

Notverkauf: wegen Systemaufgabe für A. 1000, 1-MB-Erweiterung mit Uhr, NP 750 DM für 500 DM zu verkaufen. Erweiterbar auf 4 MB. Tel. 0431/711302 ab 18 Uhr

Verkaufe Amiga 1000 ohne Monitor, dt. Tastatur 512 KB. Anruf bitte Do.-Mo. ab 20.45 Uhr. Tel. 06122/51780

512K-RAM-Erweiterung für Amiga 500 mit Echtzeit-Uhr zu verkaufen. Fast neu, nur 240 DM. Tel. ab 18 Uhr oder Sa./So.: 02254/81210

Suche defekte Hardware (Kauf möglich). Schnelle Abnahme: Melden bei R-OCCO. Tel. 04194/511

Original Wiesemann Interface 92008 zum Anschluß von C64/C128-kompatibler Peripherie (MPS 802 etc.) an den Amiga für nur 100 DM zu verk.! Tel. 0711/7280411

Olympia-Carrera am Amiga? Kein Problem! Schnittstellenbox (Original-Olympia) für Parallelport zu verkaufen. 150 DM inkl. Plan f. Verb.Kabel. 04101/46217 n. 18.30 Uhr

RAM-Chips, 256 KBit, Typ 81256-12 (120 ns), 32 Stück zum Preis v. 5 DM/St. zu verkaufen. 04101/46217 n. 18.30

Amiga 1000-512K, deutsche Tastatur, mit zweiten Laufwerk für 1400 DM. Monitor 1081 für 500 DM verkauft: 0231/403758 (ab 16 Uhr)

Notverkauf: wegen Systemaufgabe für A. 1000, 1-MB-Erweiterung mit Uhr, NP 750 DM für 500 DM zu verkaufen. Erweiterbar bis zu 4 MB. Tel. 0431/711302 ab 18 Uhr

Neuwertige Hardware gegen Höchstgebot: Hitrans-Akustikkoppler, 300 Baud, Epson-Plotter HI-80 mit Erweiterungs-ROM FX80 und Stifte. Reiner Baumann, Schillerweg 26, 7186 Blaufelden

Verkaufe Adpaterkabel zum Anschluß einer C1541, C1570, C1571-Floppy an einen Amiga. Preis 35 DM. 3,5-Zoll/2DD leer. Disks für 20 DM (10er Pack). Tel. 02365/59829

Farbdrucker Okimate 20, IBM-Version für Amiga/Atari ST, neuwertig u. orig. verpackt incl. Farbbänder u. Papier für 350 DM. Tel. 02158/6856

Verk. Netzteil für A500 mit dreifacher Leistung des Originalnetzteils, keine Probleme bei Hardwareerw. VB 150 DM. Tel. 02851/7590 ab 17 Uhr

Verkaufe Amiga 1000 (PAL-Ver. mit dt. Tastatur) mit Basispaket + 2 Lw. (NEC 1036 a) + Maus + DPaint II + Spiele + Literatur. Meldet Euch bei Ralf ab 18 Uhr. T. 030/2516226, VB 2200 DM

Verkaufe Amiga 1000 mit Software und Originalen, Zusatzlaufwerken, Leerdisks, (10 St. für 22 DM 2DD) ca. 80 St. Wolfgang Jesberger, Schelmenweg 4, 7916 Strass

Sampler A1000 f. Audiom., Future Sound, Perf. Sound, DELA S., belegt nur Parallelp., 120 DM. Echtzeituhr A500—2000, Joy.2 durchg. 95 DM. Tel. 02261/28141 Ralf

Stop
Sidecar für Amiga 1000, voll IBM-kompatibel (z.B. Flugsimlator II) incl. 512-K-Speichererweiterung zu verk. Tel. 02385/5180

$5\frac{1}{4}$-Zoll-Laufwerk, 4 Mon. alt, 40/80 Tracks 300 DM, Disketten $3\frac{1}{2}$Zoll 2DD (10 St.) 20 DM, Disks $5\frac{1}{4}$ Zoll 2DD, 8 DM. Ernst Hecht, Postfach 1273, 8482 Neustadt, Tel. 09602/2515

Amiga 1000 + Digitizer + 2LW + Monitor 1081 + Soft. + Bücher. Tel. 02823/3141 (ab 18 Uhr)

Verkaufe Amiga 1000 ohne Monitor, dt. Tastatur 512 KB. Anruf bitte Do.—Mo. ab 20.45 Uhr. Tel. 06122/51780

Amiga Reference Manuals: ROM-Kernel Libraries and Devices, Exec, Intuition. Neupreis 225 DM für 175 DM (neuwertig). M. Vitolini, Forellenweg 11, 8210 Prien

Epson EX-800 Color, neuwertig, originalverpackt, mit Traktor, Einzelblatteinz., neue Farbb. und Par. Kabel. VB 1100 DM. M. Vitolini, Forellenweg 11, 8210 Prien

Verkaufe 768-KB-Erweiterung + Megakick (Kickstart) für 1000er. Neu, ungebraucht, nur 430 DM. D. Barth, Heppenheimer Str. 35, 6149 Fürth/Odw. 06253/21267

Amiga 1000, Frontabdeckung fehlt, 1400 DM. Mit viel Software, ohne Monitor, PAL Version, 6103 Griesheim, Schönweibergasse 74, Frank Prediger, Tel. 06155/5655

Verkaufe Amiga 1000 mit 512 KB, Amiga 1060 Sidecar mit 512 KB und Farbmonitor 1081, VB 3000 DM. Christoph Hille, Tel. 0531/77680, Adolfstr. 32, 3300 Braunschweig

Verkaufe Amiga 1000 o. Monitor incl. 2.FD-LW und Drucker-Kabel f. 1300 DM. Tel. 0211/298637 ab 19 Uhr

Amiga 1000 + 14 MHz 68020/18 MHz 68881-Karte 20/81 Karten — auch einzeln. Tel. 0721/709913

Amiga 500 inkl. Speichererweiterung 1 MB m. Uhr mit Monitor 1081 wegen A 2000 für 1700 DM. Tel. 0234/512505

Original-Software 50% unter akt. Preis. DPaint2, DigiPaint, Prism, PageSetter, Grafiktablett ProwDraw, A4, 480 DM. Tel. 0221/217384

Verkaufe Amiga 1000 m. 512 KB, Amiga 1060 Sidecar mit 512 KB und Farbmonitor 1081, VB 3000 DM, Christoph Hille, Adolfstr. 32, 3300 Braunschweig, Tel. 0531/77680

Verkaufe TA Typenraddrucker TRD 7020 für Amigas + Traktor für Endlospapier. Tel. 0911/421909 ab 19 Uhr, Möse verlangen

Verkaufe Amiga 1000 incl. Monochr.-Monitor kaum gebr. + Handb., div. Softw. (VBM-Text) VB 1350 DM. Tel. 09376/774 ab 17 Uhr

MEGAMIGA! 768 KB-Erweiterung für Amiga 1000. Endlich kommen Sie billig an die dringend nötigen 1 MB. MEGAMIGA! Nur 333 DM. Bernhard Kirsch, 0681/71796

Amiga 500 zu verkaufen. 512 KB Ramerweiterung mit Uhr, 2 orig. Spiele (Garrison, Bad Cat), ca. 6 Monate alt. Wegen A2000 für 900 DM, Tel. 06196/46575

Amiga 500 Fans! Verkaufe Inboard Speichererweiterung für Amiga 500. 1,5 MB RAM, wird im Rechner eingebaut. NP 900 DM, 2 Monate alt. VB 790 DM. Tel. 06196/46575

A1000 RAM-Erweiterung intern (CPU-Sockel 1—4 MB möglich) ohne RAMs, incl. bat + gep. Uhr, Workbench, Kickstart, techn. Unterlagen 300 DM. Tel. 04101/46217 ab 18.30 Uhr

Amiga 1000 mit 512 KByte und diverse Software. Tel. 089/223983 abends 9039458. Suche: S/W Video-Kamera für Amiga-View

Speichererweiterung für Amiga 1000 auf 1 MB. Neu! Nur 339 DM! Einbauanleitung liegt bei. Lars Blumenhofer, Dorstener Str. 31, 4350 Recklinghausen (02361/15943)

Verkaufe Amiga 1000 wegen Amiga 2000; mit 512 KB, Software, Handbücher, VB 1100 DM. Tel. 0911/761740 (ab 17 Uhr)

AMIGA Sounddigitizer, f. fast alle Samplerprg., Metallgehäuse, 59 DM. J. Weinert, Münsterberger Str. 11, 2900 Oldenburg, Tel. 0441/62741

Amiga 1000, 512 KB + Farbmon. 1081, deutsche Tastatur + Bücher + Software + Joystick 1800 DM, Sidecar m. 20 MB Filecard, Uhrenkarte, V.24, Software, 1900 DM. Tel. 0228/624256 ab 17.30 Uhr

Power without the Price — für nur 3,33 DM im Monat bist Du Mitglied im Computerclub International. Der führende Club für alle Amiga User! Info: Tel. 02361/15943

Verkaufe: Superdrucker Epson FX80 inclusive Interface bzw. Paralellkabel für sage und schreibe 600 DM. Sehr guter Zustand. NP 1500 DM. Tel. 06152/63321 ab 17 Uhr

Verkaufe 2-MB-RAM-Box (Golem) für A1000. Tel. 02823/3141 öfter versuchen!

STOP
Sidecar für Amiga 1000, voll IBM-kompatibel (z.B. Flugsimulator II), incl. 512-K-Speichererweiterung + DOS zu verkaufen. Tel. 02385/5180

Verkaufe AMIGA 1000 (dt. Tastatur u. PAL-Ver.) mit Basispaket + 2 Lw. (NEC 1036a) + Maus + DPaint II + Spiele + M&T Buch u. Zeitschriften. Für VB 2200 DM bei Ralf ab 18 Uhr, Tel. 030/251/6226

Verkaufe Amiga 1000 mit 3,5 und 5,25 Zoll Laufwerken. Dazu Disks bespielt mit neuerer Software und Originale! Wolfgang Jesberger, Schelmenweg 4, 7916 Straß

Ausland

Verkaufe wegen Systemwechsel: Amiga 1000 + Monitor 1081 (mit 512 KB), Top-Zustand + viel Software aus allen Bereichen. 1900 Sfr. Felix Kägi, CH-061/475793

Schweiz! 3,5- und 5,25-Zoll-Disketten. Name oder No Name, Disk.boxen, Joysticks und MCs zu Tiefstpreisen. Markus Hodel, Quaistr. 2, 6403 Küßnacht, Tel. 041/812338

Private Kleinanzeigen

Verkaufe Audio Digitizer für 100 DM. Kompatibel mit u.a. Aegis Audio Master. Neu! I. van Riemen, van Kyfhoeklaan 94, 2597 TE Den Haag, Holland. Tel. 0031/70240247

Verschiedenes

68000er
Ausgaben 1/87—8/87, wie neu (!), zu verk. Angebote (nur schriftl.) an Olaf Winkler, Uhlandstr. 7, 7889 Grenzbach-Whylen

Bücher: Data Becker C für Einst. 20 DM, Tips u. Tricks 30 DM, Supergrafik a.d. Amiga 40 DM, A. für Einsteiger 100 DM, De Luxe Grafik M&T 30 DM, ab 19 Uhr 0209/200520

Achtung! Suche Kontakte zu Amiga-Usern in der Gegend von Xanten, Alpen, Geldern. Bin Einsteiger! Tel. 02801/2936 ab 16 Uhr.

Hallo Amiga-Users in Baden-Würtemberg. Möchte Adressenbibliothek für B. U. einrichten. Schickt Eure Adresse an: A. Dreier, Lerchenstr. 20, 7272 Altensteig. Thanks

We are searching for cool Contacts. Call: 05732/3455 (Dirk). Only 18—21!

Suche Kontakte zu netten Amiga-Anwendern, auch Neulingen, zwecks Erfahrungs- und Programmtauschs. Ich nutze mein Gerät privat und geschäftlich. Michael Lange, Boppstr. 4, 1000 Berlin 61. Tel. 030/6931023

Modula-Einsteiger sucht Fortgeschrittene, die gerne Hifestellung geben. Tausche auch Spiele! Schreibt an Thomas Konrad, Saumstr. 34, 7963 Altshausen, Tel. 07584/701

Suche Anleitungen, Lösungen zu Games und Anwenderprogrammen. Schreibt an Michael Jähnel, Albrechtstr. 16, 4100 Duisburg 12

Adventures! Verkaufe orig. Shadowgate u. Jinxter. mit Lösungen! Keine Einzeltips! Tel. 02134/32855

Suche Zeitschrift Amiga 8/87, 10/87, 11/87, 12/87. Zahle 5 DM pro Heft! Bitte an Hans-Jörg Ziefle, Untere Sonnenhalde 66, 7292 Baiersbronn 1

Alfreds Computer Club
Wir suchen noch Mitglieder! Gemeinsamer Kauf/Tausch von PRGs, PD-Software etc. ACC/Rietburgstr. 2, 6708 Neuhofen

Phalanx II 20 DM, Ultima III 60 DM, 256 K RAM-Erweiterung für A1000 50 DM, Original CBM A2000 Einbaulfw. 250 DM, externes Laufwerk (NEC 1036 A) 280 DM. Daniel Hotzy, Eichhaldenweg 16, 7344 Gingen

Suche
Amiga-Magazin 6/7 und 8/9 und Literatur zum Amiga (500). Tel. 02101/467246, Bernd Pischke, Rembrandtstr. 96, 4040 Neuss

Private Kleinanzeigen

An alle einsamen Computerfans! Ich vermittle Kontakte! Infos: M. Scholz, Am Jödenkamp 14, 4840 Wiedenbrück-Oder, Tel. 05242/54241, nur von 18.30 bis 20 Uhr

Suche Kontakte zu Amiga-Usern im Raum LD zwecks Erfahrungs- und Softwaretauschs. Bitte meldet euch bei Achim Steigner, Lazarettstr. 19, 6740 Landau, Tel. 06341/31990

Original-Software 50% unter akt. Preis. DPaint2, DigiPaint, Prism, PageSetter, Grafiktablett, ProwDraw, A4, 480 DM. Tel. 0221/217384

Amiga-Beginner sucht im Raum München, der mich am Wochenende gegen Std.-Lohn am Computer einweisen kann, einen Amiga-Freak. Tel. 089/3136857 ab 17 Uhr

Löse Ihre speziellen Software-Probleme auf Amiga in C oder Basic (Gymnasiast, K12). Tel. ab 17 Uhr 08191/4923

1. Computerclub Schaffhausen sucht Mitglieder aus CH. Softwarebib., Clubzeitung usw. Info bei: T. Clemens, Hirschwiesenweg 8, 8200 Schaffhausen. Nur Schweizer!

MIDI + Homerecording-User-Club. Info gegen Rückporto von: Kay-Uwe Berghof, Roseggerstr. 5, 5600 Wuppertal 2

Du kannst ewig vor dem Amiga versauern! Oder Du wendest Dich an Deutschlands führenden Amiga-Club. Info kostenlos. CCI, Dorstener Str. 31, 4350 Recklinghausen, 02361/15943

Suche Kontakte zu A1000-Anwendern und Clubs im Raum Würzburg. Besonders an Erstellung eigener Prg. in C interes. Bernd Schulte, 8701 Theilheim, Tel. 09303/636

Amiga-Literatur: Manuals (englisch), 68000-Prozessorbuch, Assemblerbuch, C für Anfänger, MS-DOS, C-Library. Ernst Hecht, Postf. 1273, 8482 Neustadt, 09602/2515

MIDI + Homerecording-User-Club. Info gegen Rückporto von Kay-Uwe Berghof, Roseggerstr. 5, 5600 Wuppertal 2

Suche Partner für Animation, Raytracing, Sound im Raum Stuttgart. Tel. 0711/875957. N. Barthelmeß (29 J.), Markgröninger Str. 42, 7000 Stuttgart 40

Private Kleinanzeigen

Searching for new contacts on Amiga the Skull. Tel. 056621/73417

Suche Gleichgesinnte, die sich mit mir verbünden wollen und gute Kontakte zu Gruppen in aller Welt haben. Auch DFÜ. Call Darth Vader. 0043/022312850

Gewerbliche Kleinanzeigen

AMIGA-Finanzbuchhaltungssoftware
Testdiskette/Softwareprobe für 15,—
bei MICROTEC, Rigaweg 1,
3300 Braunschweig, Tel. 05309/1466.
Auch Händleranfragen erwünscht

Disketten NoName 3,5″ 1DD-DM 24,50
Disketten NoName 3,5″ 2DD-DM 26,50
AMIGAS-DRUCKER-ZUBEH.
ELEKTR. BAUTEILE
Preisliste anfordern: W. Niemann,
Walsumermarkstr. 140, 4200 Oberhausen 11

PUBLIC-DOMAIN für Amiga & IBM
Tiefstpreise + 24 Std. Versand
Katalogdisk gegen 5,— Vorkasse
Funkcenter Mitte GmbH
Klosterstr. 130, 4000 Düsseldorf 1
Tel. 0211/362522
Mailbox 0211/360106 — 18—9 Uhr

Suche Programmieranfänger (Assembler und/oder C) mehr oder weniger fortgeschritten, zum gemeinsamen Erlernen. Kann auch Tips und Ratschläge brauchen. Darth Vader, Ottakringer Str. 105, A-1160 Wien

Amiga-2000-Besitzer: Bitte schreibt mir über Hardware-Erfahrungen u. Probleme mit d. A2000 unter d. Motto »A2000-User vereinigt Euch«. Helmut Wilde, Überfuhrstr. 35—37, A-1210 Wien

Suche Amiga-Computer-Club in der Schweiz, vorzugsweise Region Zürich, Erfahrungsaustausch + Projekte z.B. Modula II. M. Oehler, Dietikon, 01/7407694

Gewerbliche Kleinanzeigen

AUGENSCHMAUS direkt von A.U.G.E. 4000, 16 PD getestet mit dt. Anleitung, Musik-, Bildern, Icon-, Fontdisks; 200 US-PD; ab 4 DM, Infodisk 5 DM. Moltkestr. 115, 4000 Düsseldorf 30

Große, europaweite Briefsimulationen! Wenn Sie einmal einen Staat im Europa NAPOLEONS regieren möchten, zu fremden Welten vorstoßen wollen, dann sollten Sie einmal Infos von den komplexesten Spielen überhaupt anfordern! DECOS, Egenolffstr. 29, 6000 Frankfurt/Main 1

Staubschutzhauben für alle Amiga Computer 24,90 DM. 512-K-RAM mit Realtime Uhr, 405 DM. Nur Versand! Selbstabholung nach Vereinbarung! Gesamtliste anfordern! Kennwort Amiga! JEPOSOFT, Kruppstr. 9, 4040 Neuss 21, Tel. 02107/12338

PD auf 3.5″ 2DD Markendisk DM 4,60—5,60 auf 5.25″ 2DD NoName 3,00—4,00; Fish, Auge Panorama, Faug, ACS, Amicus, u.a.; ca. 400 PD! Je 10 Stück 10 Pf. Abzug! 2 Infodisks 5,—
VERBATIM Leerdiskette 3,5″ 2DD 2,90 DM 5.25″ 2DD 1,40; Dt. Anl. Bal. of Power 15,— Porto/Vers. frei ab 200,— BW sonst 5,— NN Th. Broschard, Eulerstr. 10, 6806 Viernheim, Tel. 06204/2988, tagsüber 06331/43104

* Amiga-Software von AMIGAWARE *
* AW-Faktura (deutsch) nur 99,— DM *
* PD Soft ab 3,50 DM, Info gratis bei *
* AW, Herbert Blöhm, 8391 Schlinding 7 *
* Tel. 08544/481 (! nur 19—20 Uhr!) *

Public-Domain: Fish, Faug, Auge, Panorama, Tornado, ACS, BCS, TBAG, Kickstart, Amicus, Chiron, — alle Disk mit »Qualitätsgarantie« — ab 3,50 DM/Disk (Staffelpreise) 2 ausführliche Katalogdisk — 6 DM DONAU-SOFT. Maik Hauer, Postfach 1401, 8858 Neuburg, 08431/49798

Amiga Public-Domain Katalog: PD-Soft, Pf. 359, 4290 Bocholt

Ausland

X-tension Peripherie
EDV-Beratung T. Eder
Ihr Partner in Sachen
Hardware-Software-Systemberatung
Tel. 0911/329558 17—20 Uhr

DEUTSCHE ANLEITUNGEN!!
INTERESSE??
Dann anrufen 06103/22477 oder schreiben
H. Reinhardt, 607 Langen, Margaretenstr. 18

Public Domain (Fish u.a.) pro Disk 3,50 DM; Info gegen Rückporto. P. Keim, 5000 Köln 30, Vogelsanger Str. 34, Tel. 0221/520765

GRATISLISTE über AMIGA-Public-Domain bei F. Neuper, 8473 Pfreimd, Postfach 72, anfo.

Wichtiger Hinweis:

Zur Bezahlung von Kleinanzeigen werden weiterhin keine Briefmarken angenommen

C-Kurs für

In diesem letzten Kursteil unseres C-Kurses für Einsteiger erfahren Sie alles über den C-Preprozessor und dessen Arbeitsweise. Die Beschreibung vieler systemunabhängiger Funktionen rundet den Kurs ab. Aber keine Angst, ein neuer C-Kurs folgt demnächst.

Vor der eigentlichen Übersetzung einer C-Quelldatei wird, in der Regel vom C-Compiler, der Preprozessor aufgerufen. Dieser bereitet das Quellprogramm zur Übersetzung vor. Vor dem ersten Durchlauf des C-Compilers durchsucht er das gesamte Quellprogramm nach Zeilen, die mit dem Doppelkreuz (#) beginnen, denn dort erwartet er seine Anweisungen. Wir können also festhalten: alle Zeilen, die Preprozessoranweisungen enthalten, beginnen mit einem Doppelkreuz. Die uns bereits bekannte #define-Anweisung, mit deren Hilfe man symbolische Konstanten definiert, ist, wie Sie richtig vermuten, eine solche Preprozessoranweisung. Eine Aufgabe des Preprozessors oder Precompilers, wie er manchmal genannt wird, ist es, diese symbolischen Konstanten durch ihren tatsächlichen Wert zu ersetzen. Darüber hinaus erledigt er aber noch folgendes:

```
- Kommentare aus der Quelldatei entfernen;
- Makros expandieren;
- Include-Dateien, sogenannte Headerfiles, einfügen;
- Steuerung der bedingten Compilierung;
```

Teil 6

KURSÜBERSICHT

Unser C-Kurs soll allen Lesern einen Einstieg in die Programmiersprache C bieten. Kenntnisse in anderen Sprachen sind nützlich, aber nicht notwendig. In den einzelnen Kursteilen finden Sie folgende Themen:

TEIL 1: Geschichte von C; das Modulkonzept; Beispielprogramme mit der Funktion »printf«, Syntaxdiagramme

TEIL 2: Einfache Datentypen, aufgeteilt nach Konstanten und Variablen

TEIL 3: Zusammengesetzte Datentypen; Zeiger; Typ-Konvertierung und Operatoren

TEIL 4: Ausdrücke und Anweisungen mit erklärenden Beispielen

TEIL 5: Aufbau von Funktionen; Parameterübergabe und Aufruf mit Programmen

TEIL 6: Der C-Preprozessor und die Bibliotheksfunktionen mit Beispielen und Übungen

Dem Programmierer stehen folgende preprozessorspezifische Kommandos zur Verfügung: Zur Definition symbolischer Konstanten und Makros »#define« und »#undef«, zum Einfügen der Headerfiles oder Include-Dateien »#include« und zur bedingten Compilierung »#if«, »#ifdef«, »#ifndef«, »#else« und »#endif«. Eine Übersicht über die allgemeinen Preprozessoranweisungen, welche übrigens nicht Bestandteil der Sprache C sind, finden Sie in Tabelle 1. Manche Preprozessoren verstehen aber noch weitere Kommandos wie zum Beispiel »#asm« und »#endasm«, auf die wir aber leider in diesem Kursteil nicht mehr eingehen können. Eine Beschreibung dieser speziellen Kommandos entnehmen Sie bitte dem Handbuch des entsprechenden Entwicklerpakets. Wir besprechen im folgenden die allgemeinen Anweisungen, die in jedem Preprozessor enthalten sein sollten und die oben genannten Aufgaben erfüllen.

Headerfiles oder Include-Dateien

Schon mehrfach wurde erwähnt, daß ein C-Programm aus mehreren Quelldateien bestehen darf. Dies hat den Vorteil, daß Sie immer wiederkehrende Funktionen nicht immer neu schreiben müssen. Die einzelnen Quelldateien bleiben klein und überschaubar. Um eine Datei aus Ihrer Bibliothek einem Quellprogramm einzubinden, müssen Sie dem Preprozessor nur deren Namen mitteilen. Eine ganze Bibliotheksdatei wird durch eine einzige Zeile im Quellprogramm repräsentiert. Die Zeile

```
#include "stdio.h"
```

wird vom Preprozessor durch die komplette Datei »stdio.h« ersetzt. Eine Bibliotheksdatei kann beispielsweise Funktions-, Variablen- oder Makrodefinitionen, aber auch weitere #include-Zeilen beinhalten. In dem Unterverzeichnis »INCLUDE« der Systemdiskette Ihres C-Compilers finden Sie zahlreiche Headerfiles. Schauen Sie sich doch einmal einige davon an! Sie werden dort viele Beispiele für Aufrufe der verschiedensten Preprozessoranweisungen entdecken, aber auch interessante Strukturen und Makros können untersucht werden. Die Headerfiles, die in C-Programme eingebunden werden können, enden zur Unterscheidung von anderen Files mit ».h«. Steht der Dateiname hinter der #include-Anweisung in Anführungszeichen, so wird das entsprechende File zuerst im aktuellen Verzeichnis gesucht. Falls es dort nicht gefunden wurde, wird in anderen, vorher festgelegten Katalogen weitergesucht. Wurde das Headerfile aber in Kleiner-/Größerzeichen als »Klammern« angegeben, etwa so:

```
#include <stdio.h>
```

so unterbleibt die Suche im aktuellen Verzeichnis; der Preprozessor sucht das File sofort in dem vorher vereinbarten Verzeichnis, beispielsweise im Verzeichnis »INCLUDE«. Zeilen, welche die Preprozessoranweisung #include enthalten, stehen gewöhnlich direkt am Anfang einer Quelldatei (siehe hierzu auch Tabelle 2 — der Aufbau eines C-Programms). Es werden oft sogar mehrere Bibliotheksroutinen aufgerufen.

Symbolische Konstanten

Mit diesem Thema haben wir uns ja bereits im Abschnitt »Daten« auseinandergesetzt. Hier noch einmal ein Beispiel:

```
#define PI 3.14
```

Beachten Sie bitte, daß das erste Zeichen der Zeile das Doppelkreuz ist. Zeilen, die Preprozessoranweisungen enthalten, beginnen immer mit dem Doppelkreuz. Alles was hinter dem Namen, in unserem Beispiel »PI«, steht, ersetzt nach dem Preprozessordurchlauf den Namen selbst. Wenn Sie also die Zeile in unserem Beispiel mit einem Semikolon

#define	symbolische Konstante oder Makro definieren
#undef	löschen einer symbolischen Konstanten oder eines Makros
#include	Header- oder Includefile einfügen
#if	Test einer Bedingung für die bedingte Compilierung
#ifdef	Test eines mit #define definierten Namens
#ifndef	Test eines mit #define definierten Namens
#else	Einleitung des Alternativ-Zweigs bei der bedingten Compilierung

Tabelle 1. Die Preprozessor-Anweisungen mit Bedeutung

Anfänger

abschließen, wozu man als C-Programmierer immer verleitet wird, ersetzt der Preprozessor brav alle Namen »PI« im Quellcode durch »3.14;«! Wird hinter dem Namen kein Wert angegeben, so wird für diese symbolische Konstante der Wert 1 festgelegt. Diese beiden Preprozessoranweisungen haben demnach die gleiche Wirkung:

```
100 #define AMIGA
100 #define AMIGA 1
```

Sie dürfen natürlich jeden Namen nur ein einziges Mal definieren, deshalb wurde den beiden Zeilen auch die gleiche Zeilennummer vorangestellt. Die #define-Anweisung darf, unter Berücksichtigung der genannten Regeln, überall im Quelltext stehen. Im allgemeinen werden symbolische Konstanten und Makros aber, wie Tabelle 2 verdeutlicht, gleich nach den #include-Anweisungen, also am Anfang des Quelltextes vereinbart. Es ist gebräuchlich und sinnvoll, den Namen einer symbolischen Konstanten in Großbuchstaben anzugeben, um sie von Variablennamen gut unterscheiden zu können.

Symbolische Konstanten »leben« zwischen den Anweisungen »#define« und »#undef«, wie statisch definierte Variablen. Allerdings dürfen sie nicht redefiniert werden. Nach einer #undef-Anweisung, wie zum Beispiel

```
#undef PI
```

gelten symbolische Konstanten als nicht definiert. Man könnte nun den somit freigewordenen Namen für eine erneute Definition verwenden; davon raten wir Ihnen aber, der Lesbarkeit und Nachvollziehbarkeit Ihres Quellprogramms zuliebe, dringend ab.

- Programmkopf als Kommentar, mit Name, Versionsnummer, Datum und Autor
- #include-Anweisungen
- #define-Anweisungen
- externe Definitionen von Funktionen
- externe Definition von Variablen
- Vorwärtsdeklarationen von Funktionen
- statische Definitionen von Variablen
- Funktionsdefinitionen

Tabelle 2. Der allgemeine Aufbau eines C-Programms. Die einzelnen Punkte sollten wegen der Übersichtlichkeit mit einigen Leerzeilen optisch getrennt werden.

Die Makrodefinition

Weiter oben wurde beschrieben, daß alles, was bei einer #define-Anweisung dem Namen folgt, diesen nach dem Preprozessordurchlauf ersetzt. Diese Tatsache können wir uns zunutze machen und immer wiederkehrende Ausdrücke oder Anweisungen abkürzen. Ein Beispiel hierfür ist der Zugriff auf eine Strukturkomponente im Programm »NDir.c« aus dem letzten Kursteil:

```
#define TYPE eintrag->
fib_DirEntryType
```

Anstelle eines langen Ausdrucks muß nur noch TYPE geschrieben werden. Der Quellcode wird kompakter, übersichtlicher und leichter lesbar, die Gefahr von Tippfehlern wird gemindert. Wenn Ihnen eine Zeile zur Makrodefinition nicht ausreicht, können Sie das Zeilenende mit dem Fluchtsymbol »\« unterdrükken. Manchmal ist das nötig, denn wir können noch einen Schritt weitergehen und sogar kleinere »Funktionen« einem einzigen Namen zuweisen. Bei der Makrodefinition darf dem Namen noch ein formaler Parameter in runden Klammern folgen. Als Beispiel ein Makro namens QUAD:

```
#define QUAD(zahl) zahl
*zahl
```

Hinter der runden Klammer wird festgelegt, wie der Parameter zahl verarbeitet werden soll. Ein Aufruf wie »QUAD(5)« wird vor der Übersetzung des Quellprogramms durch »5*5« ersetzt. Man könnte Makrodefinitionen mit Funktionsdefinitionen vergleichen. Der Makroname mit Klammer und Parameter ist der Funktionskopf, die Festlegung der Operation der Funktionsblock. Makros sind jedoch im Vergleich zu Funktionen schneller in der Ausführung, da ja weder Aufruf noch Rücksprung erfolgen muß. Darüber hinaus sind sie vom Variablentyp unabhängig; sie funktionieren mit allen skalaren Datentypen. Der Nachteil der Makrodefinition ist, daß ein Makro nicht wie ein Unterprogramm aufgerufen, sondern an den Ort des Aufrufs kopiert wird, was eine Vergrößerung des Objektcodes (Objectcode = das übersetzte Programm) zur Folge hat. Makronamen sollte man, zur besseren Unterscheidung von Funktionsnamen, in Großbuchstaben schreiben. Gerade bei der Definition von Makros kommt es oft zu unerwünschten Effekten. Schauen Sie sich unser Beispiel noch einmal an: Der Aufruf »QUAD(10)« wird ersetzt durch »10*10« - ergibt 100. Ist vollkommen richtig, aber was ergibt QUAD(9+1)? Dieser, dem ersten scheinbar gleichen Aufruf wird übersetzt zu »9+1*9+1« und liefert als Ergebnis den Wert 19! Auch hier gilt: besser zwei Klammern zuviel als eine zuwenig! Richtig und sicher definiert sieht unser Makro dann so aus:

```
#define QUAD(zahl)
((zahl)*(zahl))
```

Aber auch beim Makroaufruf lauern die Fehlerteufel: Der Aufruf QUAD(x++) wird ersetzt durch (x++)*(x++)! Die Variable x wird also zweimal inkrementiert! Prüfen Sie deshalb vor der Anwendung eines Makros den Ersatztext und den Aufruf doppelt genau. Die Zeit, die Sie hier »opfern«, ist verschwindend gering im Vergleich zu der Zeit, die Sie bei der Suche eines solchen Fehlers verlieren; von der nervlichen Belastung ganz zu schweigen. Die Preprozessoranweisung »#undef QUAD« löscht die Definition des Makros QUAD.

Bedingte Compilierung

Bedingte Compilierung bedeutet, daß Teile einer Quelldatei nur unter bestimmten Bedingungen übersetzt werden. Die Bedingung kann zum Beispiel der Wert einer Konstanten oder eines konstanten Ausdrucks sein:

```
#if KONSTANTE
```

Die dieser Anweisung folgenden Zeilen werden übersetzt, wenn der (konstante) Ausdruck hinter »#if« einen Wert ungleich 0 besitzt. Man kann die Compilierung bestimmter Programmteile auch von der Definition eines Namens abhängig machen:

```
#ifdef NAME
```

Eine Übersetzung erfolgt nur, wenn der Name hinter »#ifdef« in einer #define-Anweisung definiert wurde. Hier die Umkehrung der »#ifdef«-Anweisung:

```
#ifndef NAME
```

Der Abschnitt nach einer solchen Anweisung wird nur dann compiliert, wenn der Name hinter »#ifndef« nicht definiert ist. Wenn die Bedingung der genannten Entscheidungsanweisungen erfüllt ist, werden alle folgenden Zeilen bis zu der Preprozessoranweisung »endif« ausgeführt.

```
100 ...
110 #ifndef ABS
120 #define ABS(zahl)
((zahl)>=0 ? (zahl) :
-(zahl))
130 #endif
140 ...
```

In diesem Beispiel wird, falls der Name ABS noch nicht definiert ist, das Makro ABS vereinbart. Es ermittelt den Absolutwert einer beliebigen Zahl. Der Aufruf dieses Makros könnte so aussehen:

```
200 int a;
210 ...
220 printf("Der Absolut
wert von %d ist %d!\n",
a,ABS(a));
230 ...
```

Am Ende der Anweisungsfolge zwischen der Entscheidungsanweisung und #endif darf noch ein Alternativblock angeboten werden, welcher mit der Preprozessoranweisung »#else« eingeleitet wird. Er wird übersetzt, wenn die Bedingung der Entscheidung nicht zutrifft. Ein Beispiel:

```
100 ...
110 #define AMIGA
120
130 #ifdef AMIGA 140
/* Routinen für AMIGA */
150     ...
160     ...
```

```
170 #else
180      /* Routinen für
XXX-Rechner */
190      ...
200      ...
210 #endif
220 ...
```

Dieser Ausschnitt eines Quellcodes könnte aus einem Programm stammen, das auf mehreren Rechnern laufen soll. Wie Sie wissen, ist ja eine der Stärken der Programmiersprache C die Portabilität der Quellcodes. Nur die systemabhängigen Routinen müssen angepaßt werden. In unserem Beispiel wird in Zeile 110 der Name AMIGA mit dem Wert 1 definiert; erinnern Sie sich: wenn die Angabe eines Wertes fehlt, wird automatisch der Wert 1 angenommen. Somit werden die Zeilen 140 bis 160, welche Amiga-spezifische Funktionen enhalten könnten, compiliert. Fehlt Zeile 110 jedoch, so werden die Zeilen 180 bis 200, passend für den Rechner XXX, übersetzt. Die systemabhängigen Funktionen sind auf Hardware und Betriebssystem eines speziellen Rechners zugeschnitten und daher auch nur auf diesem Rechner lauffähig. Es existiert aber noch eine Reihe von Funktionen, die unabhängig vom Rechnertyp ihren Dienst verrichten. Es sind dies meist grundlegende Funktionen, ohne die keine Programmiersprache auskommt. Auf den folgenden Seiten werden Ihnen die wichtigsten dieser systemunabhängigen Routinen vorgestellt. Manchmal werden Informationen benötigt, die im Headerfile »stdio.h« festgelegt sind. Es ist daher nötig, dieses, wie weiter oben beschrieben, im Quellcode einzubinden (#include <stdio.h>). Die speziellen Funktionen, die uns der Amiga zur Verfügung stellt, werden zu einem späteren Zeitpunkt, in einem Kurs für fortgeschrittene C-Programmierer, vorgestellt.

Die folgenden Funktionen beziehen sich auf das aktuelle Ein-/Ausgabegerät (Standard Ein- und Ausgabe oder stdin/stdout). In aller Regel wird dies der Bildschirm als Ausgabegerät und die Tastatur als Eingabegerät sein. Das in »stdio.h« definierte Makro »getchar()« dient dazu, ein einzelnes Zeichen von stdin zu lesen. Hier der Aufruf und die aktuellen Parameter dieser Funktion:

```
100 char a;
110 a=getchar();
```

c	Ausgabe eines Zeichens
d	Ausgabe einer Dezimalzahl
e	Ausgabe einer Floatzahl in wissenschaftlicher Notation
f	Ausgabe einer Floatzahl
g	Ausgabe einer Floatzahl (kürzeste Darstellung bei voller Genauigkeit)
o	Ausgabe einer Oktalzahl
s	Ausgabe eines Strings
u	Ausgabe einer vorzeichenlosen ganzzahligen Dezimalzahl
x	Ausgabe einer Hexadezimalzahl
l	Ausgabe von d, o, u oder x als long
ZAHL	bestimmt die Anzahl der auszugebenden Zeichen
ZAHL.	Anzahl der Zeichen vor dem Dezimalpunkt
.ZAHL	Anzahl der Zeichen nach dem Dezimalpunkt
%	Ausgabe eines Prozentzeichens
—	(Minuszeichen) linksbündige Ausgabe
*	nächsten ganzzahligen Wert der Argumentliste ausgeben.

Tabelle 3. Diese Möglichkeiten bietet die Funktion printf().

Ein- und Ausgabefunktionen

Der Zeichenvariablen wird der return-Wert der Funktion getchar() zugewiesen, es ist dies das von stdin gelesene Zeichen. Das Gegenstück zu getchar() ist putchar(), mit deren Hilfe ein einzelnes Zeichen auf stdout ausgegeben werden kann. Dieses Makro benötigt als Parameter das auszugebende Zeichen.

```
100 char a;
110 putchar(a);
```

Während getchar() und putchar() nur einzelne Zeichen verarbeiten, können mit gets() und puts() Strings, also Zeichenketten, eingelesen beziehungsweise ausgegeben werden. Da Zeichenketten, als zusammengesetzte Variablen, nicht als aktuelle Parameter dienen können, muß ein Zeiger auf die Adresse des ersten Zeichens im Speicher übergeben werden.

```
100 char *string;
110
120 gets(string);
130 puts(string);
```

Mit gets() werden alle Zeichen bis zum Return eingelesen. Das Returnzeichen wird gegen ein Nullbyte (\0) ausgetauscht und die ganze Zeichenkette ab der angegebenen Adresse im Speicher abgelegt. Die Funktion puts() gibt alle Zeichen ab einer bestimmten Adresse, in unserem Beispiel der Adresse »string«, bis zum Nullbyte aus. Anstelle dieses Nullbytes wird ein Return (\n) ausgegeben. Die Funktion printf(), zur formatierten Ausgabe von Zeichen auf das aktuelle Ausgabegerät, haben Sie bereits kennengelernt. Hier noch einmal die Syntax:

```
100 char *kontroll
string;
110 printf(kontroll
string,argumente);
```

Das erste Argument setzt sich aus den auszugebenden Zeichen und den sogenannten Formatangaben zusammen. Die Formatangaben werden selbst nicht ausgegeben, sondern dafür die Werte der dem Kontrollstring folgenden Argumente. Diese werden gemäß der Formatangabe konvertiert und an deren Stelle ausgegeben. Die Formatangabe wird mit dem Prozentzeichen (%) eingeleitet. Tabelle 3 zeigt die Zeichen, die diesem folgen dürfen und die entsprechende Bedeutung. Auch bei der Funktion scanf(), mit deren Hilfe man Zeichen von stdin lesen kann, wird mit solchen Formatangaben gearbeitet (Tabelle 4). Diese sind den Formaten von printf() sehr ähnlich, haben sogar oft die gleiche Bedeutung. Die Anzahl der Zeichen, die gelesen werden sollen, also die Länge des Eingabefeldes, kann, wie Sie aus Tabelle 4 entnehmen können, frei gewählt werden. Wurde die Länge nicht definiert, so werden alle Zeichen bis zum ersten Space-, Tabulator- oder Returnzeichen gelesen.

```
100 char *a;
110 int *b;
120
130 scanf("%c%d",a,b);
```

Das erste Argument ist der Kontrollstring. Es sollen in Zeile 130 ein Zeichen und eine Dezimalzahl von der Tastatur gelesen werden. Die entsprechenden Werte werden in den Adressen, auf die die beiden Zeiger a und b verweisen, gespeichert.

Das Arbeiten mit Zeichenketten

Die Programmiersprache C unterstützt das Arbeiten mit Zeichenketten nicht. Befehle, wie sie in Basic gebräuchlich sind, wie zum Beispiel MID$, VAL oder STRING$, wird man vergeblich suchen. In jeder Funktionsbibliothek warten aber eine Fülle von entsprechenden C-Funktionen auf ihren Einsatz. Einige davon sollen hier vorgestellt werden. Beginnen wir mit einer Funktion, die wir bereits in einem früheren Kursteil kennengelernt haben, sscanf():

```
100 char *string, *kon
trollstring;
110 ...
120 sscanf(string,kon
trollstring,zeiger);
```

Die Funktion sscanf(), welche eng mit scanf() verwandt ist, liest Zeichen aus einer Zeichenkette. Das erste Argument von sscanf() ist der String, aus dem gelesen werden soll. Die beiden folgenden sind mit den Argumenten von scanf() identisch. Es handelt sich um den Kontrollstring mit den Formatanweisungen und den Zeiger auf die Zielvariablen. Auch die Funktion printf() hat einen Zwil-

c	ein Zeichen wird erwartet
d	eine Dezimalzahl wird erwartet
D	eine long-Dezimalzahl wird erwartet
f	eine Floatzahl wird erwartet
F oder lf	eine double-Zahl wird erwartet
o	eine Oktalzahl wird erwartet
O	eine long-Oktalzahl wird erwartet
s	ein String wird erwartet
x	eine Hexadezimalzahl wird erwartet
X	eine long-Hexzahl wird erwartet
ZAHL	bestimmt die Eingabefeldlänge
*	Eingabefeld überspringen

Tabelle 4. Die Formatangaben der Funktion scanf() ähneln denen von printf()

AMIGA ★ Public Domain Software ★ MS-DOS

Über **600 Disketten** lieferbar: **Fish** 1-134, **Panorama** 1-56, **Faug** 1-53, **Amicus** 1-22, **Auge** 1-15, **Taifun** 1-50, **ES-Soft** 1-55, Reiner Wolf **RW-Disks 1-30**, **Chiron Conceptions** 1-40, **ACS** 1-23, **Tornado-Spez.** 1-30, **Kickstart 1-57**, **TBAG** 1-7, **SACC** 1-4, Casa Mi Amiga sowie Winners Cycle System, Amuse, Amigazin, Juice Magazin, AAA und viele andere Disks mehr.

Einzeldisk	DM 4,90
bis 10 Stück	DM 4,85
bis 30 Stück	DM 4,80
bis 60 Stück	DM 4,70
bis 90 Stück	DM 4,60
bis 120 Stück	DM 4,50
bis 150 Stück	DM 4,40

auf 3,5″-Disketten 2DD.

Nur **145,–** für jedes Paket mit **30 PD-Disketten**, inkl. Porto, Verpackung und **CLI-Hilfe** DirUtil, bei Vorkasse (V-Scheck oder bar). Zum Beispiel:

Paket Nr.	1a	= Fred Fish	Nr.	1 - 30
Paket Nr.	1b	= Fred Fish	Nr.	31 - 60
Paket Nr.	1c	= Fred Fish	Nr.	61 - 90
Paket Nr.	1d	= Fred Fish	Nr.	91 - 120
Paket Nr.	**3**	**= Panorama**	**Nr.**	**1 - 30**
Paket Nr.	4	= Faug Hot Mix	Nr.	1 - 30
Paket Nr.	7	= Kickstart	Nr.	1 - 30
Paket Nr.	**8**	**= Taifun**	**Nr.**	**1 - 30**
Paket Nr.	9a	= ES-Soft	Nr.	1 - 30
Paket Nr.	9b	= ES-Soft	Nr.	31 - 60
Paket Nr.	10	= Chiron Conc.	Nr.	1 - 30
Paket Nr.	**11**	**= Tornado-Spez.**	**Nr.**	**1 - 30**

Oder Sie stellen sich **Ihr ganz persönliches Paket** aus unserem Amiga PD-Katalog zusammen.

2 Katalog-Disketten mit Information über Inhalt der Programme für Amiga 500/1000/2000 gegen DM 5,– in Briefmarken/bar/V-Scheck anfordern!

Spezial-Katalog über Original PC-SIG-Public Domain- & Shareware-Programme für den Amiga mit **PC-Karte** oder mit **MS-DOS-Transformer** gegen DM 5,– in Briefmarken/bar/V-Scheck anfordern!

Am gleichen Tag des Bestelleingangs erfolgt der Versand unserer Kataloge!

Versandkosten PD-Disketten ...

Porto für Inland/Ausland	DM 3,–
Nachnahme für Inland	DM 4,–
Nachnahme für Ausland	DM 14,–

Achtung neu! Ray-Tracing-Construction-Set V2.0, siehe Amiga 1.88, S. 117. **Komplettpaket** 3 Programmdisks & 2 Katalogdisks & ausgedruckte deutsche Anleitung für **DM 29,95 inkl. Porto.**

Achtung neu! Bei Abnahme ab 30 Disketten **kostenlos** für den Anfänger oder Profi ... **CLI-Hilfe** auf Diskette, lesen, kopieren, editieren, sortieren, drucken, renamen und vieles mehr, ähnlich wie **CLIMATE** oder **ZING** ... --- DirUtil IV.12 ---

UWE SCHMIELEWSKI

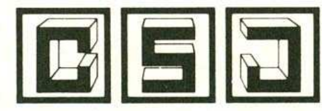
- - Ihr Public-Domain-Archiv für Amiga - -

Haroldstr. 71 · 4100 Duisburg 1 · Tel. 0203/376448
BTX *0203376448# · Fax 0203359690

Jeden Monat Software im Briefkasten!

Regelmäßig jeden Monat bekommen Sie Ihre Public-Domain-Software zugeschickt, mit den neuesten Informationen in der PD-Szene und mit einem Rabatt von 10%.

Abonnement-Preise entnehmen Sie unserem Katalog oder gegen Rückporto aus unserer Informationsmappe.



COMPUTERSOFT

AMIGA ARCADE GAMES	
BACKLASH	64,90
BAD CAT	59,90
BIG DEAL	79,90
BRAINSTORM	34,90
COGANS RUN	49,90
DEFENDER OF THE CROWN	86,00
FEUD	34,90
GARRISON II	69,90
INSANITY FLIGHT	79,90
INTO THE EAGLES NEST	64,90
LEVIATHAN	64,90
MERCENARY COMPENDIUM	79,90
MIKE — THE MAGIC DRAGON	29,95
MOEBIUS	79,90
PHALANX II	29,95
PHANTASIE III	59,00
PINBALL WIZARD	49,95
ROADWARS	64,90
SINBAD	86,00

AMIGA SPORT GAMES	
GRAND SLAM TENNIS	94,90
GRID START	34,90
INDOOR SPORTS	79,90
KARATE KID II	79,90
SOCCER KING	29,95
TESTDRIVE	98,90
THAI BOXING	34,90
WINTER OLYMPIADE 88	64,90

AMIGA ADVENTURE	
BUREAUCRACY	84,90
DARK CARSTELL	79,90
GNOME RANGER	49,90
HELLOWOON	74,90
JINXTER	79,90
KINGS QUEST I + II + III	79,90
KNIGHT ORC	59,90
LEISURESUIT LARRY	64,90
THE PAWN	79,90

HITS ★ HITS ★ HITS

JAGD AUF ROT. OKTOBER	**74,90**	**KING OF CHICAGO**	**99,90**
LAS VEGAS	**34,90**	**BALANCE OF POWER**	**89,90**
CRAZY CARS	**49,90**	**WESTERN GAMES**	**59,00**

SPACEPORT	64,90
STARGLIDER	79,90
STRIP POKER	29,95
TERRAMAX	64,90
TERRORPODS	79,90
THUNDERBOY	59,90
TOLTEKA	64,90
TRIVIA PROBE (DEUTSCH)	34,90
XR 35	34,90

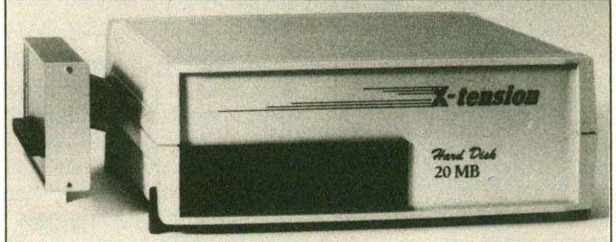
AMIGA-STRATEGIE	
KAMPFGRUPPE	89,00
OGRE	79,90
ROADWAR 2000	79,90
ROADWAR EUROPA	79,90
TETRIS	64,90

AMIGA-ANWENDER	
AEGIS AUDIOMASTER	129,90
AEGIS DRAW PLUS	498,00
AEGIS SONIX VERSION 2.0	159,00
AEGIS VIDEO TITLE	359,00
AMIGA TOOLS	49,95
DELUXE PAINT II (DEUTSCH)	298,00
DIGI PAINT (DEUTSCH)	169,00
DRUCKERANPASSUNG CP-80X	59,90
LOGISTIX (DEUTSCH)	399,90
MAUSE MATTE	19,90
SCULPT 3D PAL-VERSION	229,00
SILVER	349,90
SOUND SAMPLER A500/2000	149,90
THE 64 EMULATOR A1000	169,90
THE 64 EMULATOR 500/2000	169,90

★ Wir suchen noch Programmautoren ★ ★ Preisänderungen vorbehalten ★

CSJ COMPUTERSOFT GmbH ★ HÄNDLERANFRAGEN ERWÜNSCHT ★
An der Tiefenriede 27 ★ 3000 Hannover 1 ★ Tel. Bestellservice (0511) 886383
sofort **CSJ NEWS** anfordern (Computertyp ang. u. Briefmarken 1,50 DM beilegen)
Versand Inland: Vorkasse + 3 DM (Euroscheck in DM): per Nachnahme + 7,– DM

AMIGA™ "Fast"-Disk



Blitzschnell durch Interleavefaktor 0

- z.B. AMIGA-"Assem" (ca. 64 KB) wird in weniger als 1,5 Sec. geladen.
- Anschlußfertig mit einer Speicherkapazität von 20 – 72 MB
- Einfache Software-Installation auf der Bootdiskette
- Demo-Diskette mit Installationsbeispielen
- ½ Jahr Garantie

Professionelles Arbeiten zu attraktiven Preisen:

20 MB DM 1498.–
40 MB DM 1998.–
72 MB DM 2998.–

Außerdem bei uns zu bestellen:

AMIGOS FÜR AMIGA™

Anschlußfertige Laufwerke: 3½" DM 319.–
5¼" DM 379.–

Beachten Sie auch den ausführlichen Testbericht in dieser Ausgabe



Wünschen Sie weitere Informationen rufen Sie uns an!
Tel. 0911/32 77 32 · Postfach 84 00 73 · 8500 Nürnberg 84

F01: 3½-Zoll-Slimline, beige
intelligente Abschaltung,
MS-DOS-kompatibel
DM 289,–

F03: 5¼-Zoll-Slimline, beige
intelligente Abschaltung,
MS-DOS-kompatibel, 40/80 Tr.
DM 389,–

A01: Amprom
Eprommer 8-64 KByte
DM 259,–

GANZMETALL-GEHÄUSE

Floppy-Bus durchgeführt

Amiga-Floppys
ab **239,–** DM
MS-DOS-kompatibel für
3½″ und 5¼″

vts data GmbH

Postf. 1110, 5014 Kerpen 1
Postf. 400621, 5 Köln 40, Tel. 02234/71601

Fremdlaufwerke am AMIGA

Sie kennen das Problem: Sie wollen sich auf ein Programm konzentrieren. Im Fremdlaufwerk liegt keine Diskette. Das ständige »klack, klack, klack ...« stört Ihre Konzentration.

Nicht so mit ONEX

Sie arbeiten mit dem Amiga-Transformer unter MS-DOS. Eine Diskette soll vom Fremdlaufwerk eingelesen werden. »Read Error Track 35«, oder gar »Track 17«.

Nicht so mit ONEX

Sie laden ein Programm mit viel Speicherplatzbedarf. Ihr Fremdlaufwerk ist aktiv und raubt dem Rechner die benötigten 23 KByte, weil Sie es nicht abgeschaltet haben: »Failed Return Code 123« oder »Not enough Memory« sind die Folge.

Nicht so mit ONEX

ONEX-Zweitlaufwerke verfügen über eine intelligente Schaltelektronik, die Ihnen hilft, die genannten Probleme zu beherrschen. Eine Änderung auf Ihren Disketten ist nicht erforderlich.
ONEX-Zweitlaufwerke für Commodore Amiga sind Slimline Low-Power-Laufwerke bekannter Marken-Hersteller.

ling, der die Arbeit mit Zeichenketten erleichtert. Die Funktion sprintf() erlaubt die formatierte Ausgabe auf einen String:

```
100 char *string;
110 sprintf(string,"%d"
,100);
```

Entsprechend der Formatangabe werden Argumente konvertiert und in einem String abgelegt. In unserem Beispiel wird die Dezimalzahl 100 als die Zeichenkette »100\0« ab der in »string« gespeicherten Adresse gespeichert. Die Funktion sprintf() arbeitet genau wie printf(), die Ausgabe erfolgt jedoch nicht auf stdout, sondern in den als erstes Argument genannten String. Mit Hilfe dieser Funktion können etwa Integerwerte in Zeichenketten umgewandelt werden. Eine andere Art der Umwandlung bieten die »ato« Funktionen. Dieses Kürzel steht für »ASCII to«. Wir unterscheiden drei Fälle:

```
100 char *string;
110 double f_zahl;
120 int i_zahl;
130 long l_zahl;
140 ...
150 atof(string);
160 atoi(string);
170 atol(string);
```

Die Funktion atof() wandelt von ASCII nach float und liefert einen double-Wert zurück, atoi() und atol() wandeln nach int beziehungsweise long und liefern ebensolche Werte.

Umwandlung möglich

Manchmal findet man auch die Umkehr dieser Funktionen, zum Beispiel itoa() oder ftoa(). Einen Vergleich zweier Zeichenketten miteinander ermöglicht uns strcmp():

```
100 char *a, *b;
110 strcmp(a,b);
```

Wenn beide Strings gleich sind, liefert strcmp() den Wert null. Ist der erste der beiden lexikographisch größer, so wird ein Wert größer null, im anderen Fall ein Wert kleiner null, zurückgegeben. Wenn nur eine bestimmte Anzahl von Zeichen verglichen werden sollen, hilft uns strncmp() weiter:

```
100 char *a, *b;
110 int anzahl;
120
130 strncmp(a,b,anzahl);
```

Der Aufruf von strncmp() ist dem von strcmp() ähnlich, es muß nur noch ein dritter Parameter, die Anzahl der zu vergleichenden Zeichen, angegeben werden. Wir können auch einen String nach einem einzelnen Zeichen durchsuchen. Einen Zeiger auf das erste gefundene Zeichen von links liefert index(), während rindex() einen Zeiger auf das erste gefundene Zeichen von rechts zurückgibt. Wenn das betreffende Zeichen in der Zeichenkette nicht vorkommt, reichen beide Funktionen einen Nullzeiger zurück.

```
100 char *string;
110 char zeichen;
120
130 index(string,zei
chen);
140 rindex(string,':');
```

In Zeile 130 wird ein String von links nach rechts nach einem Zeichen durchsucht. In der Zeile darunter wird ein String von rechts nach links nach einem Doppelpunkt durchsucht. Das Kopieren von Zeichenketten von einem Speicherbereich in einen anderen geschieht mit der Funktion strcpy():

```
100 char *nach, *von;
110 int anzahl;
120 ...
130 strcpy(nach,von);
140 strncpy(nach,von,an
zahl);
```

In Zeile 130 werden alle Zeichen ab der Adresse »von« zu der Adresse »nach« kopiert. Der Aufruf von strncpy() zeigt, wie man eine bestimmte Anzahl von Zeichen ab Adresse »von« zur Adresse »nach« überträgt. Auch ein Aneinanderhängen von Zeichenketten ist möglich:

```
100 char *string1,
*string2;
110 int anzahl;
120 ...
130 strcat(string1,
string2);
140 strncat(string1,
string2, anzahl);
```

Die Zeichen ab der Adresse von »string2« werden im Speicher hinter die Zeichen von »string1« angehängt. Das erste Zeichen von »string2« ersetzt das Nullbyte von »string1«. Der Funktionsaufruf in Zeile 140 zeigt, daß dies auch mit einer beliebigen Anzahl von Zeichen ab »string2« realisierbar ist. Mit der letzten der hier beschriebenen String-Funktionen können Sie die Länge, sprich die Anzahl der Zeichen einer Zeichenkette ermitteln. Das Nullbyte, das jede Zeichenkette abschließt, wird nicht berücksichtigt. Hier ein Beispiel für die Funktion strlen():

```
100 char *string;
110 int anzahl;
120 ...
130 anzahl=strlen
(string);
```

Aus dem Aufruf in Zeile 130 geht hervor, daß strlen() eine Integerzahl, eben die Anzahl der Zeichen zurückgibt.

Ein-/Ausgabe im Dateisystem

Nachdem Sie die verschiedensten Möglichkeiten zur Verarbeitung von Daten unter C kennengelernt haben, soll jetzt noch besprochen werden, wie Sie diese auf Diskette sichern und später auch wieder einlesen können. Es stehen hierfür zwei Arten von Funktionen zur Verfügung: Funktionen, die mit einem Datei-Deskriptor arbeiten, und solche, die auf einen FILE-Pointer verweisen. Beginnen wir mit den Funktionen, die mit dem Deskriptor arbeiten. Ein Dateideskriptor ist einfach eine ganze Zahl, die bei jedem Öffnen einer Datei hochgezählt wird. Sie dient als Index, wenn diese Dateien bearbeitet werden sollen. Sie erhalten den Deskriptor mit der Funktion open():

```
100 char *filename;
110 int deskriptor, mo
dus;
120
130 deskriptor=open(fi
lename,modus);
```

Eine Datei wird geöffnet. Die Funktion gibt, wenn sie erfolgreich arbeiten konnte, den Dateideskriptor zurück, ansonsten einen Wert kleiner null. Sie können zwischen verschiedenen Modi wählen, und zwar bedeutet modus=0, daß die Datei zum Lesen geöffnet werden soll; modus=1, die Datei soll beschrieben werden; modus=2, die Datei wurde zum Lesen und Schreiben geöffnet. Gleich das Komplement zu open(): die Funktion close() schließt eine Datei.

```
100 int deskriptor, er
ror;
110 ...
120 error=close(deskrip
tor);
```

Der aktuelle Parameter dieser Funktion ist der Deskriptor, den man über die Funktion open() erhalten hat. Die Variable error erhält den Wert null, wenn die Datei geschlossen wurde, und einen Wert kleiner null, wenn ein Fehler auftrat. In der geöffneten Datei kann mit der Funktion read() gelesen werden:

```
100 char *buffer;
110 int deskriptor, an
zahl, bytes;
120 ...
130 bytes=read(deskrip
tor,buffer,anzahl);
```

Neben dem Dateideskriptor muß die Adresse eines Puffers, der die gelesenen Daten aufnimmt, und die maximale Anzahl der zu lesenden Zeichen übergeben werden. Die Funktion gibt als return-Wert die Anzahl der tatsächlich gelesenen Zeichen zurück. Die maximale Anzahl von Zeichen kann sich von der tatsächlichen Anzahl unterscheiden, wenn zum Beispiel das Dateiende eines Files erreicht wird, bevor alle Zeichen eingelesen werden konnten. Wenn beim Lesen der Datei ein Fehler stattgefunden hat, gibt read() einen Wert kleiner null zurück. Mit den gleichen aktuellen Parametern wird die Funktion write() aufgerufen. Diese Funktion ermöglicht uns, wie bereits ihr Name vermuten läßt, in eine Datei zu schreiben.

```
130 bytes=write(deskrip
tor,buffer,anzahl);
```

In die Datei, welche mit »deskriptor« indiziert ist, werden »anzahl« Bytes, die sich im Speicher ab der Adresse »buffer« befinden, geschrieben. Die Funktion write() gibt, falls sie fehlerlos arbeiten konnte, die Anzahl der tatsächlich geschriebenen Zeichen zurück, andernfalls einen Wert kleiner null. Wenden wir uns hiermit den Funktionen zu, die mit dem FILE-Pointer arbeiten. FILE ist eine Strukturvariable, die im Headerfile stdio.h definiert ist. Einige Funktionen werden Ihnen bekannt vorkommen, es sind dies unter anderem Funktionen, die printf() und scanf() ähnlich sind. Doch bevor diese zum Einsatz kommen können, muß die Datei noch geöffnet werden:

```
100 char *filename, mo
dus;
110 FILE file;
120 ...
130 file=fopen(filena
me,modus);
```

Das erste Argument der Funktion fopen() ist ein Zeiger auf den Filenamen. Es schließt sich der Modus an. Es sind wie bei open() mehrere Modi möglich. Hier werden diese jedoch

nicht mit einer Zahl, sondern mit dem Zeiger auf eine Zeichenkette dargestellt:

```
"r" - Datei zum Lesen
(read) öffnen
"w" - Datei zum Schrei
ben (write) öffnen
"a" - Datei zum Anhängen
(append) öffnen
```

Viele C-Compiler bieten noch weitere Möglichkeiten; read() und write() sind sicher die wichtigsten. Nach der Ausführung gibt fopen() einen Zeiger auf die entsprechende FILE-Struktur zurück. Falls jedoch ein Fehler auftrat, wird ein Nullbyte geliefert. Eine bereits geöffnete Datei kann mit der Funktion fclose() geschlossen werden. Als Argument dient der Zeiger der FILE-Struktur.

```
100 FILE *file;
110 ...
120 fclose(file);
```

Auch bei den folgenden Funktionen muß dieser FILE-Pointer immer angegeben werden, zum Beispiel bei der Funktion zum Lesen des nächsten Einzelzeichens einer Datei, fgetc():

```
100 FILE *file;
110 char zeichen, re
turn;
120 ...
130 zeichen=fgetc(file);
140 ...
150 return=fputc(zei
chen,file);
```

Funktionen mit Rückgabewert

fgetc() liefert das gelesene Zeichen zurück. Mit fputc() wird ein einzelnes Zeichen in eine Datei geschrieben. Das Zeichen selbst wird als return-Wert zurückgegeben. Tritt ein Fehler auf, so geben beide Funktionen EOF (End Of File = -1) zurück, eine symbolische Konstante, die in stdio.h definiert ist. Die Funktionen getw() und putw() funktionieren genau wie fgetc() und fputc(), nur wird ein Wort (16 Bit) gelesen beziehungsweise geschrieben. Weiter existiert in dieser Reihe eine Funktion fgets(), die einen String einliest, und fputs(), die einen String speichert.

```
100 char *buffer, *re
turn;
110 FILE *file;
120 int anzahl;
130 ...
140 return=fgets(buffer,
anzahl,file);
```

getchar()	Zeichen lesen
putchar()	Zeichen schreiben
gets()	String lesen
puts()	String ausgeben
scanf()	formatierte Eingabe aus stdin
printf()	formatierte Ausgabe auf stdout
sscanf()	Lesen aus einem String
sprintf()	In einen String schreiben
atof()	String nach double wandeln
atoi()	String nach int wandeln
atol()	String nach long wandeln
strcmp()	Zeichenketten vergleichen
strncmp()	wie oben, nur bestimmte Anzahl von Zeichen
index()	String von links nach rechts nach einem Zeichen durchsuchen
rindex()	wie oben, aber von rechts nach links
strcpy()	String kopieren
strncpy()	wie oben, nur bestimmte Anzahl von Zeichen
strcat()	Strings aneinanderhängen
strncat()	wie oben, nur bestimmte Anzahl von Zeichen
strlen()	Ermitteln der Anzahl der Zeichen in einem String
open()	Datei öffnen (Deskriptor)
close()	Datei schließen (Deskriptor)
read()	Daten lesen (Deskriptor)
write()	Daten speichern (Deskriptor)
fopen()	Datei öffnen
fclose()	Datei schließen
fgetc()	Zeichen lesen
fputc()	Zeichen speichern
getw()	Wort lesen
putw()	Wort ausgeben
fgets()	String lesen
fputs()	String ausgeben
fprintf()	formatierte Ausgabe in eine Datei
fscanf()	formatierte Eingabe aus einer Datei
sin()	Sinus
sinh()	Sinus hyperbolicus
asin()	Arcus Sinus (-PI/2 bis PI/2)
cos()	Cosinus
cosh()	Cosinus hyperbolicus
acos()	Arcus Cosinus (0 bis PI)
tan()	Tangens
tanh()	Tangens hyperbolicus
atan()	Arcus Tangens (-PI/2 bis PI/2)
atan2()	Arcus Tangens (-PI bis PI)
exp()	Exponentialfunktion
log()	Logarithmus naturalis
log10()	Zehner-(Dekadischer-) Logarithmus
pow()	Potenzrechnung (Argument1 hoch Argument2)
sqrt()	Quadratwurzel

Tabelle 5. Eine Übersicht der in diesem Kursteil besprochenen systemunabhängigen C-Funktionen.

```
150 return=fputs(buffer,
file);
```

In Zeile 140 werden »anzahl«-1 Bytes ab der Speicheradresse »buffer« in die Datei mit dem FILE-Pointer »file« geschrieben. Die Variable anzahl beschreibt die Anzahl der Zeichen des Strings einschließlich des Nullbytes. Die Funktion fputs() legt alle Zeichen, die im Rechner ab der Adresse »buffer« bis zum Nullbyte gespeichert sind, in einer Datei ab. Das Nullbyte selbst wird nicht mitgespeichert. Beide Funktionen geben ihr erstes Argument zurück oder, im Fehlerfalle, EOF. Die beiden Funktionen zur dateibezogenen formatierten Ein- und Ausgabe, sind den bereits besprochenen Funktionen printf() und scanf() oder sprintf() und sscanf() fast gleich. Der einzige Unterschied ist, daß das erste Argument ein Zeiger auf die FILE-Struktur der entsprechenden Ein- oder Ausgabedatei sein muß:

```
100 FILE *file;
110 char *zeiger;
120 ...
130 zeiger="Datei";
140 fprintf(file,"Ausga
be auf eine %s",zeiger);
150 fscanf(file,
"%s",zeiger);
```

Zeile 140 zeigt, wie in eine Datei geschrieben wird, eine Zeile tiefer wird mittels fscanf() eine Information aus einer Datei gelesen und ab der Adresse »zeiger« im Arbeitsspeicher des Rechners abgelegt.

Alle mathematischen Funktionen arbeiten mit doppelter Genauigkeit. Es können daher nur Variablen des Datentyps »double« übergeben werden. Binden Sie bitte, mit Hilfe der #include-Anweisung, das Headerfile math.h mit ein und achten Sie darauf, daß Sie die Mathematik-Bibliothek mitlinken! Die Syntax der mathematischen Funktionen ist immer gleich und daher schnell beschrieben:

```
100 double ergebnis,a,b;
110 ...
120 ergebnis=mathe_funk-
tion(a);
```

Die einzige Ausnahme bildet die Funktion pow(). Diese wird mit zwei Argumenten aufgerufen:

```
120 ergebnis=pow(a,b);
```

Die Variable ergebnis erhält den Wert a hoch b. Ergebnis und Argument(e) sind bei allen Mathematikfunktionen vom Typ double. In Tabelle 5 sind neben den Mathematikfunktionen noch einmal alle in diesem Kurs besprochenen systemunabhängigen Funktionen und deren Bedeutung zusammengefaßt. Den Abschluß soll die Funktion exit() bilden, mit deren Hilfe nicht nur ein C-Programm, sondern diesmal sogar ein ganzer C-Kurs, beendet wird. Was uns bleibt, ist Ihnen viel Erfolg und vor allen Dingen viel Spaß bei der C-Programmierung zu wünschen. Wenn Ihnen der ein oder andere Punkt noch nicht hundertprozentig klar sein sollte — sind Sie bitte nicht enttäuscht, auch in C gilt: nur die Übung macht den Meister.

(Arno Gölzer/rb)

In diesem Kurs haben Sie die Grundlagen der Sprache C kennengelernt. Sie sind jetzt in der Lage, einfache Programme auf dem Amiga zu schreiben. Aber um diesen tollen Computer wirklich auszunützen, reicht das bisher Gelernte noch nicht. Deswegen wird in einer der nächsten Ausgaben ein Kurs starten, der sich näher mit dem Amiga beschäftigt. Die Hauptthemen sind Intuition, Amiga-DOS und Grafik. Die Benutzung der Betriebssystemroutinen bildet dabei den wichtigsten Punkt. Nach diesem Kurs haben Sie Ihren Amiga mit C dann voll im Griff.



Modula-2: Der Kurs stimmt

Die Programmiersprache Modula-2 schickt sich an, auf dem Amiga ein ernsthafter Konkurrent für »C« und »Basic« zu werden. Mit dem Einsteiger-Kurs in der AMIGA und einem preisgünstigen Compiler auf Public Domain können Sie sofort in diese moderne und leistungsstarke Sprache einsteigen.

Im Jahre 1969 betrat zum ersten Mal ein Mensch den Mond. Damit begann ein neues Zeitalter in der Geschichte der Raumfahrt. Im Verlauf der folgenden Jahre wurden weitere Mondflüge unternommen, Raumsonden zum Mars und zur Venus geschickt. 1985 startete gar die europäische Raumsonde Giotto ihren Flug zum Halley'schen Kometen.

Doch noch auf anderen wissenschaftlichen Gebieten werden Fortschritte erzielt — weniger spektakulär, oft in der Stille — die ebenso bedeutend sind. Dazu gehört die Entwicklung der Programmiersprachen. Die Zentren der Forschung auf diesem Gebiet reichen von Oxford und Cambridge über Japan, die Vereinigten Staaten, bis nach Zürich an die Eidgenössische Technische Hochschule:

Im Jahre 1969 entwickelte Niklaus K. Wirth die Sprache »Pascal«. Damit begann das Zeitalter der strukturierten Programmierung. Im Verlauf der folgenden Jahre entstanden Sprachen wie »C« und »Modula«. 1980 stellte N. Wirth »Modula-2« vor — eine Kombination der Sprachen Pascal und Modula.

Pascal war ursprünglich für Lehrzwecke vorgesehen. Wirths Studenten sollten ihren Programmen Struktur und Übersicht verleihen. Bald stellte sich heraus, daß Pascal sich auch für die professionelle Programmierung in Softwarelabors eignet, ja dafür prädestiniert ist. Kein Wunder, denn Pascal ist eine strukturierte Programmiersprache: Sie erlaubt es, größere Projekte durch die Zerlegung in Prozeduren überschaubar zu machen und effizienter testen zu können. Pascal trat gegen Ende der 70er Jahre einen kaum erwarteten Siegeszug durch die Entwicklungsabteilungen der Softwareproduzenten an.

...aber ein großer Schritt...

Einige Compiler-Hersteller erkannten die Chance und brachten jeweils »ihren« Pascal-Compiler heraus (was ein Compiler ist, erfahren Sie etwas später im Text). Jeder dieser Compiler besaß anfangs seine »Eigenheiten«, die meist den Zweck hatten, kleine Schwächen der Sprachdefinition von Pascal auszubügeln. Mit der Zeit kristallisierten sich aus den unterschiedlichen »Dialekten« einige Standards, wie zum Beispiel »UCSD-Pascal« oder »Turbo-Pascal« heraus. Die Tatsache, daß sich mehrere Standards bildeten, war jedoch ein Grund vieler Softwareproduzenten, sich anderen Sprachen, zum Beispiel C, zuzuwenden. Ein weiterer Grund waren Schwächen der Sprache, die sich besonders bei der Programmierung großer Aufgaben auswirken.

Durch die Abkehr großer Softwarehäuser geriet Pascal jedoch nicht in Vergessenheit — ein Schicksal, das einigen anderen Programmiersprachen leider widerfuhr — sondern setzte seinen Siegeszug durch die Zimmer der Hobbyprogrammierer fort. So sind mittlerweile für den C 64 einige vielversprechende Pascal-Compiler auf dem Markt. Deshalb an dieser Stelle eine frohe Botschaft für alle, die bereits mit Pascal zu tun hatten: Sie werden sich beim Erlernen von Modula-2 leicht tun. Doch Pascal ist nur einer der Vorfahren zu Modula-2.

Die Sprache Modula schuf Professor Wirth, um parallele Prozesse zu studieren und zu programmieren. Er versuchte, eine Sprache zu entwickeln, die »Multitasking« unterstützt. Des weiteren integrierte er in Modula das Modulkonzept. Aus der Kombination beider Sprachen entstand schließlich Modula-2.

Modula-2 besitzt alle Vorzüge von Pascal und vermeidet dessen Nachteile. Gleichzeitig stellt die Sprache vollkommen neue Konzepte zur Verfügung, zum Beispiel:
— Modularisierung,
— interne/externe Module,
— Prozedurtypen,
— Coroutinen,
— opake Typen.

Kopplung erfolgreich

Die Neuerungen erleichtern nicht nur die Programmierung, sie erhöhen die Übersichtlichkeit sowohl kleiner und einfacher als auch — und in besonderem Maße — großer und umfangreicher Programme.

Vielleicht haben Sie bereits versucht ein C-Listing nachzuvollziehen? Sobald der C-Programmierer nicht hundertprozentig »sauber« arbeitet und seine Programme dokumentiert, wird nur er seine Werke vollkommen verstehen. Ein »Programm-« oder »Hauptmodul« in Modula-2 ist durchweg besser zu lesen und zu verstehen. Dies resultiert aus den eindeutigen und aussagekräftigen Schlüsselworten und den besser gegliederten Deklarations- und Anweisungsblöcken.

Auf die meisten der Sprachkonzepte werden wir in diesem Kurs eingehen. Vorweg sei jedoch angemerkt, daß wir nicht alles bis ins letzte Detail verfolgen werden. Hierzu wird ein Modula-2-Kurs für Fortgeschrittene folgen. Interessierte Leser, die sich schon vorher intensiv mit Modula-2, speziell auf dem Amiga, beschäftigen möchten, finden auf Seite 103 eine Liste empfehlenswerter Literatur.

Damit Sie auch gleich in die Praxis einsteigen können, ohne Ihren Geldbeutel zu belasten, haben wir uns etwas einfallen lassen. Auf der Programmservice-Diskette dieser Ausgabe finden Sie ein fast vollständiges Modula-2-System. Es handelt sich dabei um die Public Domain-Version des »M2Amiga« von A.+L. Meier-Vogt. Sie finden das Paket auch auf der Fish-Disk 113. Es besteht aus einem Editor, Compiler, Linker und einigen zur Programmierung erforderlichen Bibliotheksmodulen. Der abgespeckte Compiler kann zwar nur Programme bis zu einer bestimmten Größe compilieren und die Anzahl der Bibliotheksmodule ist begrenzt, doch zumindest für die ersten Teile des Kurses reicht das aus. Einen Test des kompletten

»M2Amiga« finden Sie auf Seite 114. Eine Kurzbeschreibung des PD-Systems konnten Sie bereits in Heft 3/88, Seite 37, entdecken.

Da die PD-Version — wir wollen sie »M2Amiga-PD taufen — die Grundlagen für den Kurs bildet, stellen wir die Komponenten genauer vor. Zuvor sollten Sie jedoch folgende Schritte nachvollziehen, um das Modula-2-System auf einer Diskette zu installieren:

1. Booten Sie Ihren Amiga mit einer eigenen Workbench.
2. Starten Sie das Programm »DoMe« auf der Programmservice-Diskette.
3. Befolgen Sie die darin gegebenen Anweisungen.
4. Machen Sie die neu erstellte Diskette mit »INSTALL« bootfähig. Sollten Sie mit dem letzten Schritt Schwierigkeiten haben, sehen Sie bitte im AmigaDOS-Handbuch oder in der AMIGA 3/88, Seite 98, in den »Tips und Tricks« nach.

Nun besitzen Sie eine Diskette, auf der sich alle benötigten Dateien befinden. Mit dieser starten Sie Ihren Amiga. Ändern Sie die »Startup-Sequence« Ihren Wünschen entsprechend, ergänzen Sie eventuell im C-Directory noch einige Befehle und schauen Sie sich an, was M2Amiga-PD bietet:

— Zunächst ist da der Editor. Ein Editor ist ein Programm, ähnlich einer Textverarbeitung. Mit seiner Hilfe tippen Sie eigene Programme in den Computer ein. Sie kennen das von Basic, wo Sie zum Beispiel beim Amiga-Basic im LIST-Fenster das Programm eingeben und hinterher mit RUN starten. Arbeiten Sie jedoch mit einem Compiler — ob in Modula-2, C oder »RzwoDzwei« — so muß der Programmtext, auch Quell-Text genannt, zunächst in einer ASCII-Datei vorliegen. Ein Editor ist also ein Hilfsmittel zur Texteingabe. Der Editor »m2emacs«, der dem M2Amiga beiliegt, stammt von dem bekannten »emacs« ab. Letzterer ist auf fast jedem Computer zu finden. Schauen Sie mal auf Ihrer Extras-Diskette nach. Dort befindet sich eventuell auch eine Version.

»m2emacs« ist die speziell auf die Programmierung mit Modula-2 zugeschnittene Fassung. Das Besondere an m2emacs ist, daß Sie eine Liste aller beim Compilieren aufgetretenen Fehler laden können und der Editor anhand dieser Liste die einzelnen Fehler im Quell-Text direkt anspringt. Sie können den Editor auf zwei Wegen starten:

1. Der Aufruf über das CLI lautet:

```
m2emacs TextName
```

Wenn Sie beim Aufruf einen Dateinamen angeben, wird nicht nur m2emacs geladen, sondern auch — sofern vorhanden — die bezeichnete Textdatei.

2. Wie alle Komponenten von M2Amiga läßt sich der Editor auch von der Workbench starten, indem Sie ihn mit dem Mauszeiger zweimal anklicken. Wenn Sie zuerst das Piktogramm einer ASCII-Datei anwählen und danach bei gedrückter <SHIFT>-Taste den Editor aufrufen, wird ebenfalls beim Starten des Editors der gekennzeichnete Text mitgeladen und auf dem Bildschirm angezeigt.

In Tabelle 1 finden Sie wichtige Editorbefehle, die Sie mit der Maus und/oder über Tastatur aufrufen. An dieser Stelle sollen nur die für die Arbeit unbedingt wichtigen Funktionen erklärt werden, damit Sie bei der Quelltexteingabe nicht auf Schwierigkeiten stoßen:

— Einen Text laden Sie mit Hilfe der Maus über das Menü »Projekt«. Wählen Sie dort »Open«, erscheint in der untersten Zeile des Editorfensters die Meldung »Öffne Datei :«, die Sie mit dem Namen der zu ladenden Datei beantworten. Befand sich vor dem Aufruf dieser Option bereits ein geänderter Text im Speicher (zu erkennen an dem Stern ganz unten links), erscheint die Frage: »Änderungen vergessen[j/n]?«, die Sie entsprechend Ihren Wünschen beantworten.

— Zum Speichern eines Textes existieren drei Optionen:

1. »Sichern« speichert den momentan sichtbaren Text unter dem in der Statuszeile angegebenen Dateinamen.
2. »Sichern als« fragt nach einem neuen Dateinamen, der anschließend in die Statuszeile übernommen wird.
3. »Sichern und Ende« arbeitet genau wie »Sichern«, beendet jedoch danach den Editor und kehrt ins CLI beziehungsweise zur Workbench zurück.

Befehl	Menü-Punkt	Wirkung
<CTRL-@>	rechte Maustaste)	Setze Marke
<CTRL-A>		Springe an Zeilenanfang
<CTRL-C>		Starte neues CLI
<CTRL-E>		Springe an Zeilenende
<CTRL-K>		Lösche bis Zeilenende
<CTRL-R>		Suche rückwärts
<CTRL-S>		Suche vorwärts
<CTRL-V>		Gehe eine Seite vorwärts
<CTRL-W>	Edit/Ausschneiden	Lösche Block
<CTRL-Y>	Edit/Einsetzen	Füge Block ein
<CTRL-Z>	Projekt/Sichern und Ende	Sichern und Ende
<CTRL-X> <CTRL-C>	Projekt/Ende	Verlasse den Editor
<CTRL-X> <CTRL-D>		Lösche aktuelle Zeile
<CTRL-X> <DEL>		Lösche aktuelle Zeile
<CTRL-X> <CTRL-M>	Modula-2/Nächster Fehler	Zeige nächsten Fehler
<CTRL-X> <CTRL-R>	Projekt/Öffne Datei	Öffne Datei
<CTRL-X> <CTRL-S>	Projekt/Sichern	Sichere aktuellen Buffer
<CTRL-X> <CTRL-V>	Projekt/Lies Datei	Lies Datei
<ESC> <.>		Setze Marke
<ESC> <B>		Gehe ein Wort zurück
<ESC> <D>		Lösche ein Wort vorwärts
<ESC> <F>		Gehe ein Wort vorwärts
<ESC> <W>		Bereich kopieren
<ESC> <Y>		Bereich auf Datei kopieren
<ESC> <%>		Ersetzen

Tabelle 1. Wichtige Editorbefehle, die dazugehörigen Menüpunkte (sofern vorhanden) und ihre Funktion

Die Lenkung

Die Schreibmarke, den »Cursor« auf dem Bildschirm, bewegen Sie mit den Pfeiltasten auf der Tastatur. In Kombination mit der Taste <ALT> dienen die Tasten zum wort- beziehungsweise seitenweisen Springen im Text. Drücken Sie die Cursortasten zusammen mit <SHIFT>, gelangen Sie an den Anfang/das Ende einer Zeile (<Pfeil links>/<Pfeil rechts>) beziehungsweise des Textes (<Pfeil oben>/<Pfeil unten>).

»m2emacs« verwaltet auf Wunsch mehrere Texte gleichzeitig. Zu diesem Zweck existieren die »Buffer«. Ein Buffer enthält immer einen Text:

Über »Lies Datei« wird beim Laden eines Textes ein neuer Buffer angelegt. Dessen Name erscheint links neben dem Dateinamen in der Statuszeile.

Mit »Neuer Buffer« wählen Sie zwischen den einzelnen Buffern (und damit zwischen den einzelnen Quelltexten). Sie können auch mittels der Blockbefehle Textbereiche von einem Buffer in einen anderen übertragen.

»Lösche Buffer« löscht einen zu bestimmenden Buffer.

Besonders bei der Erstellung von Programmen ist es nützlich, einen ganzen Textbereich löschen, kopieren oder verschieben zu können. Dazu dienen die »Blockoperationen«. Der Textbereich wird mit der Maus markiert. Dazu fahren Sie mit dem Mauszeiger auf das erste Zeichen des zu markierenden Textbereiches und drücken die rechte Maustaste. Unterhalb der Statuszeile erscheint die Meldung »Marke gesetzt«. Danach bewegen Sie den Mauszeiger rechts neben das letzte Zeichen des gewünschten Blocks und drükken die linke Maustaste. Jetzt wählen Sie eine der Blockoperationen des »Edit«-Menüs:

— »Ausschneiden« löscht den markierten Textbereich aus dem Text und speichert ihn in einem Puffer. Es ist möglich, einen mit »Ausschneiden« gelöschten Text wieder einzufügen.

— »Kopieren« überträgt den markierten Bereich in einen internen Puffer, löscht ihn jedoch nicht aus dem Text.

Teil 1

KURSÜBERSICHT

Der Modula-2-Kurs ermöglicht Ihnen den Einstieg in diese neue Sprache auf dem Amiga. Der gesamte Kurs gliedert sich in acht Teile mit folgenden Themen:

TEIL 1: Arbeit mit PD-M2Amiga; Einführung in Modula-2

TEIL 2: Ausgabeprozeduren; Importliste; Deklarationsteil; Variablentypen INTEGER und CARDINAL

TEIL 3: unstrukturierte Datentypen INTEGER — CHAR; Kontrollstrukturen If und FOR

TEIL 4: unstrukturierte Datentypen, Unterbereich und Aufzählung; Kontrollstrukturen CASE und WHILE

TEIL 5: strukturierte Datentypen ARRAY — SET; Kontrollstrukturen REPEAT und LOOP

TEIL 6: Prozeduren (Aufruf, Deklaration, Prozedurtyp und Funktionsprozeduren, Standardprozeduren)

TEIL 7: lokale und externe Module

TEIL 8: Zeiger und systemnahe Programmierung

— »Einsetzen« fügt den Inhalt des internen Puffers ab der Position der Schreibmarke in den Text ein.
— »auf Datei« speichert den Inhalt des Puffers auf eine Datei.

Zwei weitere Funktionen helfen Ihnen bei der Programmierung:
— Wenn Sie innerhalb eines Textes ein Wort oder eine bestimmte Buchstabenkombination suchen, verwenden Sie den Befehl »Suchen«.
— Soll ein gefundenes Objekt durch ein anderes ersetzt werden, wählen Sie »Ersetzen«.

Damit jedoch fürs erste genug zum Thema »Editorbefehle«. Sie werden im Laufe dieses Kurses mit einigen zusätzlichen Optionen des Editors vertraut gemacht.

M2 ruft Compiler. . .

Ist der Quelltext fertig, muß er in eine dem Computer verständliche Form gebracht werden; er muß vom Compiler in den Objekt-Code übersetzt werden. Der Computer versteht nämlich nur seine ihm eigene Maschinensprache. Das compilierte Programm läuft später mit »voller« Geschwindigkeit ab. Im Gegensatz zu der Übersetzung durch einen Interpreter (Beispiel: Amiga-Basic), wo jede Zeile immer wieder ausgewertet und ausgeführt wird.

Der Compiler prüft auch den Quelltext auf Fehler. Gerade die Modula-2-Compiler gehen dabei genau vor. Ihnen entgeht kaum ein syntaktischer Fehler —logische Fehler wird selbst der beste Compiler kaum finden. Wer schon versucht hat, den Amiga mit C zu programmieren, wird in dieser Beziehung bitter enttäuscht worden sein. Ein C-Compiler läßt sich nur sporadisch herab, ab und zu den Programmierer auf ein Versäumnis hinzuweisen. Wichtige Fehler werden vom C-Compiler meist gar nicht bemerkt. Als Modula-2-Programmierer haben Sie besonders mit dem M2Amiga, der deutsche Fehlermeldungen liefert, eine große Hilfe. Er hilft Ihnen bei der Fehlersuche. Hat der Compiler einen Fehler entdeckt, erstellt er eine spezielle Datei, ein »Fehlerprotokoll«. Es wird vom Editor beim erneuten Laden des Quelltextes mit eingelesen. Über den Menüpunkt »nächster Fehler« können Sie dann von Fehler zu Fehler springen und jeden sofort korrigieren. »Lies Fehlerliste« liest die Datei mit dem Fehlerprotokoll erneut, so daß Sie jederzeit noch einmal mit der Fehlersuche beginnen können.

Sie rufen den Compiler so auf:

```
m2c Dateiname
```

Geben Sie keinen Dateinamen an, verlangt der Compiler diesen später. Geben Sie die Namen mehrerer Quelldateien jeweils durch Leerzeichen getrennt an, übersetzt »m2c« eine nach der anderen. Weitere Optionen, die Sie beim Aufruf des Compilers noch setzen können, sind für uns noch nicht wichtig und werden später erläutert.

Wenn Sie den Compiler über die Workbench aufrufen, so gelten dieselben Regeln wie für den Editor. Statt einer Datei, die vor dem Compiler durch einmaliges Klicken angewählt wird, dürfen beim Compiler mehrere angegeben werden.

M2Amiga ist ein Einpaß-Compiler. Das heißt er übersetzt den Quelltext in einem einzigen Durchgang. Das macht M2Amiga schnell. Während der Übersetzung sehen Sie im Compilerfenster, welche Dateien »Module« das Programm während seiner Arbeit benötigt. Wir werden im Verlauf des Kurses noch genau kennenlernen, welche Datei wofür zuständig ist und was unter dem inzwischen mehrmals gefallenen Begriff »Modul« zu verstehen ist.

<F1>	Datei öffnen
<F2>	Nächster Fehler
<F3>	Suche
<F6>	Gehe zu Zeile
<F10>	Lösche Zeile
<SHIFT><F1>	Sichern
<SHIFT><F2>	Fehlerliste lesen
<SHIFT><F3>	Ersetze
<SHIFT><F6>	Information
<SHIFT><F10>	Füge neue Zeile ein

Tabelle 2. Die Belegung einiger Tasten im Editor

Das letzte Glied in der Kette vom Quelltext bis zum ablauffähigen Programm ist der »Linker«. Linker heißt ins Deutsche übertragen »Binder«. Er bindet den vom Compiler erzeugten Objektcode mit Prozeduren aus externen Modulen zusammen und speichert das Ganze als ablauffähiges Programm. So können Sie in ein Programm auch Prozeduren (Unterprogramme), die in anderen Dateien (externen Modulen) enthalten sind, einbauen. Wie dies vor sich geht, werden wir noch behandeln.

Der Aufruf des Linkers geschieht ganz analog zum Aufruf des Compilers durch:

```
m2l Dateiname
```

Für den Dateinamen geben Sie jedoch nicht den Namen des Quelltextes, sondern den der vom Compiler erzeugten Objektdatei an.

. . .und Linker

Bevor wir die einzelnen Stationen der Programmentwicklung anhand eines Beispiels durchlaufen wollen, noch eine kleine Bemerkung zu den Konventionen der Namensbildung von Quelltextdateien. Es hat sich eingebürgert, den Quelltext eines Modula-2-Programmes durch die Endung ».mod« zu kennzeichnen. Findet Sie auf einer Diskette eine Datei, die mit ».mod« endet, dürfen Sie davon ausgehen, daß es sich hierbei um den Quelltext eines Modula-2-Programmes handelt. Der Compiler erzeugt aus einer solchen Datei eine Objektdatei, an deren Namen (ohne ».mod«) die Endung ».obj« gehängt wird. Der Linker generiert aus einer ».obj«-Datei das ablauffähige Programm, ohne Endung. Zu beachten ist, daß sowohl Compiler als auch Linker (beim M2Amiga) bei der Namensbildung von dem im Modulkopf (siehe unten) eingetragenen Namen ausgehen und nicht von dem Dateinamen des Quelltextes.

Nach soviel — notwendiger — Theorie geht's mit riesengroßen Schritten zur Praxis.

Anhand eines einfachen Beispiels wollen wir nun die einzelnen Phasen der Programmentwicklung — von der Eingabe des Programmtextes bis hin zum ablauffähigen Programm — durchlaufen.

Start frei

Dazu starten Sie bitte auf einem der oben beschriebenen Wege den Editor und geben folgendes ein :

```
MODULE DasErste ;
FROM Terminal IMPORT
WriteString,WriteLn ;
BEGIN
   WriteString("Das
   erste Programm des Mo
   dula-2 Kurses") ;
   WriteLn ;
END DasErste .
```

Achten Sie auf die korrekte Schreibweise der einzelnen Worte. Besonders ist zu beachten, daß in Modula-2 zwischen Groß- und Kleinschreibung unterschieden wird.

Haben Sie den Text eingegeben, so verlassen Sie den Editor über »Sichern« oder »Sichern als« und »Ende«. Nach der obigen Empfehlung, sollten Sie für den Quelltext folgenden Dateinamen wählen:

```
DasErste.mod
```

Starten Sie nun den Compiler und geben dort den Programmnamen (mit vollständigem Pfadnamen) ein und drücken Sie <RETURN>. Sollte beim Compilieren ein Fehler auftreten (Meldung des Compilers : »— Fehler gefunden«), starten Sie den Editor erneut und vergleichen den von Ihnen eingetippten Text mit dem Beispiel. Sind Sie experimentierfreudig, wählen Sie nach dem erneuten Laden des Quelltextes im Menü »Modula-2« die Option »nächster Fehler«. Der Editor führt Sie dann zum ersten Fehler. Nach Ausbessern der Fehler und erneutem Abspeichern des Programmtextes muß der Compiler abermals gestartet werden. Dieses Wechselspiel zwischen Programmeingabe, Compilierung, Korrektur und abermaliger Übersetzung wiederholt sich, bis Sie den Quelltext korrekt eingegeben haben. Doch besser jetzt schon alle Fehler erkennen, als später dem Guru in die Arme zu laufen, wie es C-Programmierern häufig geht.

Hat der Compiler seine Arbeit zufriedenstellend verrichtet, starten Sie den Linker und geben auch dort den Namen des Programmes an. Daraufhin wird er sich für — verblüffend — kurze Zeit mit Ihrem Programm beschäftigen und meldet am Ende sogar die Größe des Programmes in Bytes. Im selben Verzeichnis wie die vom Compiler erzeugte Objektdatei befindet sich anschließend das ablauffähige Programm. Starten Sie es vom CLI aus, so sollten Sie ungefähr folgendes auf dem Bildschirm sehen :

```
1> DasErste
Das erste Programm des
Modula-2 Kurses
1>
```

Haben Sie von der Workbench aus gearbeitet, rufen Sie das vom Linker erzeugte Programm über sein Icon auf. Es öffnet sich daraufhin ein Fenster, in dem der Text erscheint. Das Fenster verschwindet, sobald Sie das

AND	ARRAY	BEGIN	BY	CASE
(CODE)	CONST	DEFINITION	DIV	DO
ELSE	ELSIF	END	EXIT	EXPORT
FOR	(FORWARD)	FROM	IF	IMPLEMENTATION
IMPORT	IN	LOOP	MOD	MODULE
NOT	OF	OR	POINTER	PROCEDURE
QUALIFIED	RECORD	(REM)	REPEAT	RETURN
SET	THEN	TO	TYPE	UNTIL
VAR	WHILE	WITH		

Tabelle 3. Reservierte Wörter des Modula-2-Compilers

Schließsymbol (»ClosingGadget«) anklicken.

Experimentieren Sie ruhig mit Ihrem ersten Programm und scheuen Sie sich dabei nicht, absichtlich Fehler einzubauen, um den Compiler zu testen. Die weiteren Teile des Kurses sollten Sie aktiv anhand der Beispielprogramme mit einem Modula-2-System verfolgen.

Beim Eingeben des ersten Programms haben Sie bereits zweimal mit dem Modulkonzept von Modula-2 Bekanntschaft gemacht. Das Wort »MODULE« leitet ein »Hauptmodul« ein. Ähnliches ist Ihnen vielleicht von Pascal bekannt, wo ein Programm mit dem Schlüsselwort »PROGRAM« beginnt. Modula-2 kennt eine ganze Reihe solcher Schlüsselworte, die jeweils eine bestimmte Aktion oder Struktur einleiten. Damit Sie sehen, was in den weiteren Folgen auf Sie zukommt, finden Sie in Tabelle 3 und 4 einen Überblick

Begleiter auf der Reise

über alle reservierten Wörter und die Standardnamen von Modula-2. Letztere umfassen alle Bezeichner (Namen) für Datentypen und Standardprozeduren. Das sind Unterprogramme, die fest in der Sprache verankert sind. Bei den »reservierten Wörtern« finden Sie Befehlsworte, die beispielsweise Kontrollstrukturen wie Schleifen einleiten oder beenden.

Damit Sie den Aufbau eines Modula-2-Programmes schon ab dieser Folge kennenlernen, finden Sie in Bild 1 ein Syntaxdiagramm. Es zeichnet ein vereinfachtes Bild eines Hauptmoduls. Weitere Bestandteile werden wir später kennenlernen und die Grafik entsprechend ergänzen. Am Ende dieses Teiles finden Sie eine kleine Einführung in die richtige Auswertung solcher Syntaxdiagramme, die uns durch den ganzen Kurs begleiten werden.

Mehrmals ist bereits der Begriff »Hauptmodul« aufgetaucht. Ein Hauptmodul ist (vorläufig) für uns nichts anderes, als ein Programm. Es ist lediglich anders aufgebaut als beispielsweise ein Basic-Programm:

Auf das Schlüsselwort MODULE, das ein Hauptmodul einleitet, folgt der Name des Moduls. Der Modula-2-Programmierer kann seinen Variablen (Prozeduren, Modulen, Konstanten und so weiter) nämlich einen eigenen Namen geben. Es ist jedoch nicht erlaubt, einem selbstdefinierten Objekt den Namen eines reservierten Wortes oder eines Standardnamens zu geben.

ABS	(BITSET)	BOOLEAN	CAP	CARDINAL	CHAR	CHR
DEC	EXCL	FALSE	FLOAT	HALT	HIGH	INC
INCL	INTEGER	LONGCARD	LONGINT	LONGREAL	MAX	MIN
NIL	ODD	ORD	PROC	REAL	SIZE	TRUE
TRUNC	VAL					

Tabelle 4. Dies sind die Standardnamen von Modula-2

Auch ein Modul erhält in Modula-2 einen eigenen Namen. Dieser wird im »Modulkopf« — die Zeile, in der das Wort MODULE steht — eingetragen.

In der folgenden Zeile machen Sie abermals Gebrauch vom modularen Aufbau eines Modula-2-Programms. In der »IMPORT«-Liste werden nach »FROM« die Namen von externen Modulen angegeben. Diese enthalten zum Beispiel Prozeduren oder Konstanten, Variablen oder Datentypen. Nach »IMPORT« wird das jeweilige Objekt angegeben, das eingeführt werden soll.

Ein »externes Modul« läßt sich am ehesten mit einer »include-Datei« eines C-Compilers vergleichen, wobei letztere nicht so flexibel ist. Betrachten Sie bitte zum besseren Verständnis einmal den Sprachumfang von AmigaBasic und Modula-2 (siehe Tabellen 3 und 4). Ihnen fällt mit Sicherheit auf, daß der Basic-Interpreter mehr Befehle (von ABS bis WRITE) bereithält, als Modula-2. Bestes Beispiel sind die Unterprogramme zur Ausgabe von Zeichenketten und Zahlen. In Basic finden Sie (siehe Basic-Handbuch) die Befehle PRINT, PRINT USING, LPRINT und dergleichen. Solange Sie auch suchen, im Sprachkern von Modula-2 werden Sie keine Prozedur finden, die PRINT ähnlich ist. Um die Kompatibilität (Übertragbarkeit) und Erlernbarkeit der Sprache hoch zu halten, besitzt Modula-2 einen relativ kleinen Sprachkern. Damit Sie dennoch Ausgaben und viele andere Aufgaben erledigen können, existieren eine Reihe von Dateien — eben die externen Module — in denen zum Beispiel eine Vielzahl verschiedener Ausgabeprozeduren enthalten sind. Um diese Prozeduren in ein Programm einzubinden, werden Sie in der Importliste (durch Komma getrennt) angegeben. Die Prozeduren werden »importiert« und können im Hauptmodul verwendet werden.

Nach »FROM« steht immer der Name eines externen Moduls, damit der Compiler weiß, wo er zu suchen hat. Wir werden in einem späteren Abschnitt dieses Kurses noch ausführlich auf die Arbeit mit externen Modulen eingehen und lernen, eigene Module zu erstellen. Im zweiten Teil werden Sie eine Aufstellung über alle beim M2Amiga zugänglichen Ausgabeprozeduren finden. In unserem Beispielprogramm sind dies die Prozeduren »WriteString« und »WriteLn«:

— »WriteString« gibt die in Klammern und Anführungszeichen stehende Zeichenkette auf dem Bildschirm aus.

Viel Zeit für Experimente

— »WriteLn« erzeugt einen Zeilenvorschub, das heißt der Cursor springt an den Anfang der nächsten Zeile.

Experimentieren Sie bis zum nächsten Mal mit diesen beiden Unterprogrammen. Orientieren Sie sich dabei am Beispiel in diesem Heft.

Der ausführbare Teil eines Programms wird durch das Schlüsselwort »BEGIN« eingeleitet.

Übrigens : In Modula-2 werden alle Deklarationen, Anweisungen und Zuweisungen durch einen Strichpunkt abgeschlossen. Dadurch erkennt der Compiler, wo eine Anweisung endet. Vergessen Sie den Strichpunkt, macht Sie der Compiler auf diesen »Leichtsinn« aufmerksam. Wundern Sie sich jedoch nicht, daß beispielsweise hinter BEGIN kein Strichpunkt folgt. Das Wort BEGIN leitet einen Anweisungsblock ein und braucht deshalb nicht mit »;« abgeschlossen werden (siehe Bild 1).

Den Abschluß eines Programmes bildet das Schlüsselwort END, gefolgt vom Namen,

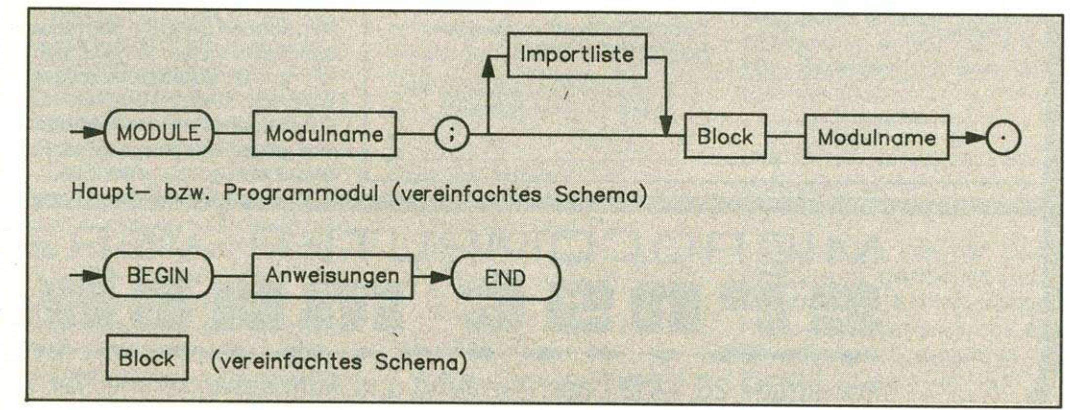


Bild 1. Dieses Diagramm zeigt den prinzipiellen Aufbau eines Programms

Viele Wege führen zu Modula-2

Damit auch Sie an dem Modula-2-Kurs in der AMIGA teilnehmen können, haben wir den Kurs so aufgebaut, daß Sie die meisten Beispiele mit dem Public Domain-Compiler auf der Fish-Disk 113 nachvollziehen können. Diese abgespeckte Fassung des kommerziell vertriebenen M2-Amiga ist für jeden Einsteiger die optimale Lösung, um sich mit der neuen Sprache anzufreunden. Besorgen Sie sich gleich eine Version:

a) Die einfachste Lösung. Fragen Sie andere Amiga-Besitzer, ob sie schon eine Version besitzen. Da es sich um Public Domain handelt, dürfen Sie sich eine Kopie machen.

b) Auf der Programmservice-Diskette zum AMIGA-Magazin werden wir ab dieser Ausgabe ständig die aktuellste Demo-Version von M2Amiga beifügen (näheres siehe Seite 153).

c) Auf der Fish-Disk 113 findet jeder Interessierte ebenfalls eine Schnupper-Packung von M2-Amiga. Diese haben wir bereits in der AMIGA 3/88 auf Seite 37 beschrieben.

d) Sie können die Demo-Diskette auch direkt bei A.+L. Meyer-Vogt bestellen. Dort bekommen Sie auch die ausgewachsene Version von M2Amiga, siehe Seite 114.

e) Wenn Sie bereits einen Modula-2-Compiler besitzen, benötigen Sie selbstverständlich nicht mehr die PD-Version.

Für welche der fünf Möglichkeiten Sie sich auch entscheiden, wir möchten, daß alle Amiga-Besitzer eine Gelegenheit bekommen, Modula-2 kennenzulernen.

der auch im Modulkopf angegeben ist. Das Zeichen ».« zeigt dem Compiler, daß an dieser Stelle das gesamte Programm beendet wird.

Damit wollen wir den ersten Teil des Kurses beschließen. Sie werden sicher eine gewisse Zeit brauchen, die vielen neuen Begriffe zu verdauen. Bis zum nächsten Mal empfehlen wir Ihnen, sich die Syntaxdiagramme und deren Beschreibung anzusehen. Gehen Sie die Diagramme anhand des Beispiels durch und vergleichen Sie die darin enthaltenen Informationen mit dem Text. Natürlich ist das Schema in Bild 1 vereinfacht. Viele Erweiterungen werden noch folgen. Wichtig ist, daß Sie sich mit dem Prinzip vertraut machen. Jeder Teil eines Modula-2-Programms läßt sich anhand eines Syntaxdiagramms darstellen und erklären. Das macht die Sache für Sie übersichtlich und die Programmierung ist nur noch ein Klacks.

So lesen Sie die Flugroute

Die Syntaxdiagramme helfen Ihnen, den Aufbau eines Modula-2-Programms zu verstehen. In Kästen mit abgerundeten Ecken oder in Kreisen stehen Sprachelemente, die für ein Programm zwingend vorgeschrieben sind: bestimmte Buchstaben, Wörter, Zeichen, Ziffern und Ziffernfolgen, zum Beispiel das erste Wort eines Moduls »MODULE«.

In rechteckigen Kästen finden Sie quasi Platzhalter, die nach der Wahl des Programmierers ersetzt werden. Der Kasten legt jeweils nur den geforderten Typ fest. (Siehe der zweite Kasten, der einen beliebigen Eintrag für den Namen eines Moduls verlangt.)

Zum Lesen der Diagramme folgen Sie der Richtung der Pfeile. An einer Verzweigung können mehrere Wege beschritten werden. Zum Beispiel in Bild 1 besteht nach dem ersten verlangten »;« die Wahl, eine Importliste einzufügen oder direkt zum Kasten mit dem Element »Block« zu gehen.

Sie werden schnell feststellen, wie leicht es ist, ein einfaches Modula-2-Programm zu schreiben. Und wenn Sie den nächsten Kursteil nicht abwarten können, experimentieren Sie mit Ihrem Compiler und verlieren Sie nicht den Mut.

(Ingolf Krüger/ub)

Vertiefende Literatur zu Modula-2:
— N. Wirth, Programmierung mit Modula-2, 3. überarbeitete Auflage, Springer-Verlag, Berlin, 1985
— I. Krüger, AMIGA Programmieren mit Modula-2, Markt&Technik Verlag, 1988, Bestellnummer MT 90554

Nüsse und Austern

In der Rubrik »Tips und Tricks« finden Sie Ratschläge aus allen Bereichen: zum Beispiel zur Programmierung, zur Anwendung und Bedienung des Amiga oder zum Anschluß eines Druckers. Suchen Sie aus, was Ihnen am besten schmeckt.

Kaffee kocht, die Luft ist stickig und der Programmierer grübelt noch bis spät in die Nacht an einer kniffligen Aufgabe. Irgend etwas läuft scheinbar noch nicht nach Wunsch. Was mag es sein, was den Übernächtigten plagt?
— Läuft eine selbstgeschriebene Routine noch nicht richtig, ist sie vielleicht zu langsam?
— Spielt der Drucker nicht mit?
— Oder versucht er aus einem Anwenderprogramm noch etwas herauszukitzeln?

Da gibt es eigentlich nur eine Hilfe. Der arme Kerl sollte einmal einen Blick in die »Tips und Tricks« werfen.

Wenn man es genau nimmt . . .

Die Funktion SQR() dient zum Berechnen der Quadratwurzel. Allerdings ist die Genauigkeit auf sechs Stellen hinter dem Komma begrenzt. Manchmal benötigen Mathematiker exaktere Werte: Weisen Sie hierzu einer Variablen mit doppelter Genauigkeit die Wurzel-Funktion zu:

```
n# = SQR(2): PRINT n#
```

Jetzt wird die Wurzel von 2 auf 16 Stellen genau ausgegeben. Dasselbe Verfahren funktioniert auch mit anderen Funktionen, zum Beispiel SIN, COS und EXP. (Volker Hauffe/ub)

Auf die Verpackung kommt es an

Stellen Sie sich einmal vor, Sie haben ein tolles Programm geschrieben und möchten es auf 3½-Zoll-Disketten verkaufen. Oder Sie möchten für Ihre Disketten ganz besondere Etiketten selber entwerfen. Doch woran hapert dieser tolle Plan meistens? Keine Druckerei ist in der Lage, Endlosetiketten in der Größe 71,5 x 69,6 zu bedrucken. Da ist Selbsthilfe angesagt: Doch ist es nicht leicht, mit einem normalen Drucker Bilder oder Schriftzüge in der richtigen Größe auf die Etiketten zu bringen. Aber zum Glück gibt es das Programm »SHOWPRINTER II« auf der Fish-Disk 90. Mit diesem Utility lassen sich Ausdrucke in jeder beliebigen Größe zu Papier bringen. Starten Sie zunächst das Programm von der Workbench. Jetzt können Sie ein beliebiges IFF-Bild laden. Mit »Print Picture« gelangen Sie ins Drucker-Menü. Die Größe des Ausdrucks bestimmen Sie durch Anklicken von »CUSTOM« und Eingabe des gewünschten Formats. Starten Sie die Ausgabe auf dem Drucker mit »PRINT«. Die exakten Werte für den Ausdruck sollten Sie durch Experimentieren selber ermitteln. Das Beispiel (Bild) wurde mit einem Epson JX-80 erstellt. (Thomas Feucht/ub)



Labels bedrucken mit »Showprinter II« ist einfach

Notepad kommt zu Ehren

Das Notepad ist eine kleine in die Workbench integrierte Textverarbeitung. Es leistet aber auch als Ersatz für den Editor gute Dienste. Wenn Sie zum Beispiel die »Startup-Sequence« ändern möchten, benutzen Sie einmal das Notepad. Wählen Sie im Menü »Projekt« den Punkt »Open«. Der Dateiname, den Sie im Requester eingeben, lautet: »df0:s/Startup-Sequence«. Eine Alternative ist es, das Notepad und den gewünschten Text direkt vom CLI aufzurufen:

```
df0:utilities/NOTEPAD df0:s/startup-sequence
```

Wer es noch bequemer haben möchte, kopiert sich das Notepad in den C-Ordner. Dann vereinfacht sich der Aufruf:

```
NOTEPAD df0:s/startup-sequence
```

Sollten Sie das Notepad dem Editor vorziehen, löschen Sie den ED ganz und benennen das Notepad mit dem Befehl RENAME in ED um. Probieren Sie einfach einmal aus, welcher Editor Ihnen besser liegt. (Michael Keukert/ub)

Textdateien in Farbe

Der TYPE-Befehl kann auch farbige, unterstrichene und kursive Buchstaben auf dem Bildschirm ausgeben. Das Programm GoAmigaDatei demonstriert dies beim Laden. Wer sich mit dem ED die Datei anschaut, die das Programm ausgibt, stellt fest, daß Escape-Sequenzen verwendet werden. Diese sind in der AMIGA (8/9)/87 Seite 88 beschrieben. Die Frage ist nun, wie kommen die ESC-Codes in die Datei? Der Editor verwendet Escape-Sequenzen als normale Kommandos. Sie können ihn also nicht verwenden, um solche Zeichen in Textdateien einzubauen. Die Sache ist recht einfach. Erstellen Sie Ihre Datei mit dem Befehl:

```
COPY * TO TestDatei
```

So leiten Sie alle Tastatureingaben auf eine Diskettendatei um. Auch die Steuercodes und sogar CTRL-Zeichenfolgen (siehe auch AMIGA 10/87, Seite 88) lassen sich speichern. Auf dem Bildschirm wird bei dieser Methode der Eingabe jeweils das Resultat des Befehls angezeigt. Das heißt, die Codes erscheinen nicht. Lassen Sie sich davon nicht irritieren. Schreiben Sie Ihren Text und schließen Sie die Datei mit <CTRL \>. Jetzt können Sie den Text mit TYPE ausgeben lassen:

```
TYPE Testdatei
```

Alle ESC- und CTRL-Folgen werden berücksichtigt. Versuchen Sie beispielsweise einmal einen Text auf rotem Hintergrund und unterstrichen schreiben zu lassen:

```
ESC  [ 3;4;43m  kursiv; unterstrichen; rot
CTRL L          löscht den Bildschirm mit der
                neuen Hintergrundfarbe
```

Auf diese Weise lassen sich mit etwas Fantasie eindrucksvolle Vorspanntitel entwerfen, die in der »Startup-Sequence« ausgegeben werden. (Axel Kaske/ub)

Starker Widerhall

Der Befehl »ECHO« hat noch einen Parameter mehr. Wenn Sie einen Text zum Beispiel sechs Zeilen tiefer als die aktuelle Cursorposition auf den Bildschirm bringen möchten, geben Sie folgendes ein:

```
ECHO "*n*n*n*n*nJetzt den Text"
```

Der Text erscheint tiefer. Der Vorteil: der ECHO-Befehl muß nur einmal geladen werden. (Norbert Cohen/ub)

Da blinkt doch was

Wenn der Amiga lange rechnet und der Bildschirm geschont werden soll, empfiehlt es sich, den Monitor auszuschalten oder die Bildschirmfarben auf Schwarz zu stellen. Daß eine Berechnung beendet ist, zeigt der Amiga, indem er die Power-LED blinken läßt. In C sieht eine solche Funktion folgendermaßen aus:

```
LED(x)          /* Funktion LED an/aus */
long x;         /* x = Blinkdauer */
{
 char *z = OxBFE001; /*Adresse des Port A CIA 1*/
 long i;
 *z = *z | 2;
 for (i=0;i<x;i++);
 *z = *z & 253;
 for (i=0; i<x;i++);

}
```

In den ersten zwei Zeilen wird die Funktion LED (x) definiert. X ist vom Typ »long«. In der vierten Zeile wird ein Zeiger auf den Port A des CIA 1 eingerichtet. Eine weitere Anwendung der Routine ist die Anzeige eines Fehlers. (Dirk Trossen/ub)

Sag mir deinen Namen

Nach langer Nacht ist das Programm endlich fertig — fast. Alles was der Programmierer noch braucht, ist eine Funktion, die durch einen bestimmten Tastendruck ausgelöst wird. »Wie war noch mal der Code von dieser..... Wo ist das Handbuch? Oje, da stehen gar nicht alle Werte und dann auch noch von einer anderen Tastatur. Verd...«

Wenn Sie alle Tastencodes Ihres Keyboards wissen möchten, schreiben Sie sich lieber eine Liste, die Sie immer verwenden können:

```
10 REM Tastenkombinationen
20 WHILE 1
30   a$=""
40   WHILE a$=""
50     a$=INKEY$
60   WEND
70   FOR I=1 TO LEN(a$)
80      PRINT ASC(MID$(A$,I,1))
90   NEXT
100 PRINT
110 WEND
```

Mit diesem Basic-Programm können Sie alle Tastatur-Codes ermitteln. Vor allem, wenn dieses verflixte Handbuch wieder weg ist. (Klaus Blaschek/ub)

Kennen Sie den schon?

Sicher haben Sie schon viel von den versteckten Witzen im Amiga-Betriebssystem gehört oder gelesen. Ein weiterer Beweis für den Humor der Entwickler ist der folgende:

Wenn Sie einmal das Pech haben, daß sich auf der Spur 40 einer Diskette ein Fehler befindet, unterziehen Sie die Diskette einer Behandlung mit dem Disketten-Doktor. Da sich auf der Spur 40 der Name der Diskette befindet, muß Amiga-DOS einen neuen Namen finden: Er lautet..... Nein, das bleibt eine Überraschung. Wenn Sie es nicht abwarten können, präparieren Sie eine Diskette, indem Sie mit einem Disketten-Monitor die Spur 40 zerstören. Und dann schauen Sie einfach mal nach, welchen Namen eine solche Diskette erhält. (Peter Eberlein/ub)

CLI-Befehle mit PLUS

Sobald Sie hinter einer CLI-Anweisung ein Plus setzen, lädt der Amiga nur den Befehl. Danach wartet er, bis Sie <Return> drücken. Erst dann führt das Betriebssystem den Auftrag durch. Ein Beispiel:

```
DIR df0: +
```

Zunächst wird der Befehl gelesen. Wechseln Sie die Disketten und betätigen <Return>. Der Amiga zeigt nun das Inhaltsverzeichnis der gerade eingelegten Diskette. Das »+« ist eine nützliche Hilfe für jeden, der nur ein Laufwerk besitzt und mit dem CLI arbeitet. (Stefan Scholl/ub)

Modular ist in

Im Zeitalter der modularen Programmierung stehen Sie bestimmt oft vor der Aufgabe, eine gute Subroutine auch in einem anderen Basic-Programm zu verwenden. Drei Wege stehen Ihnen zum Kopieren offen:

1. Abtippen ist der aufwendige Weg.
2. Auch der MERGE-Befehl erfordert einige Vorarbeiten.
3. Sichern Sie den zu kopierenden Teil mit COPY. Laden Sie das zweite Programm. An der gewünschten Stelle kopieren Sie die Routine mit PASTE. (Eike Cornelius/ub)

Spezialitäten für Schlemmer

Wissen Sie was die »Tips und Tricks« mit Nüssen und Austern gemeinsam haben? Die kleinen sind oft die besten. Doch auch die großen schmecken meist recht gut — oder?

Schreiben Sie uns einmal, welcher Trick Ihnen am besten gefallen hat. Vielleicht haben Sie eine besondere Perle gefunden.

Aber nicht vergessen, wenn Sie selbst auch ein paar Kniffe im Umgang mit dem Amiga kennen, beteiligen Sie sich an den »Tips und Tricks«. (ub)

Alternative zu MOUSE

Mit Amiga-Basic lassen sich die Koordinaten der Maus nur in Verbindung mit MOUSE(0) abfragen. Eine Alternative:

```
PEEKW(WINDOW(7)+12)
PEEKW(WINDOW(7)+14)
```

Der erste Aufruf liefert die Y-, der zweite die X-Koordinate. Die Werte beziehen sich auf den Rahmen des Fensters. Um die Koordinaten relativ zur Zeichenebene zu erhalten, müssen Sie 11 vom Y- und 3 vom X-Wert abziehen. Ein Beispiel:

```
wind& = WINDOW(7)
WHILE 1
  LOCATE 1,1
  PRINT PEEKW(wind&+14),PEEKW(wind&+12)
WEND
```

Das Programm zeigt ständig die Mauskoordinaten relativ zum Rahmen des Fensters. Wenn Sie in der ersten Zeile des Listings

```
a&=PEEKL(WINDOW(7)+46)
```

schreiben und in der vierten Zeile

```
PRINT PEEKW(a&+18),PEEKW(a&+16)
```

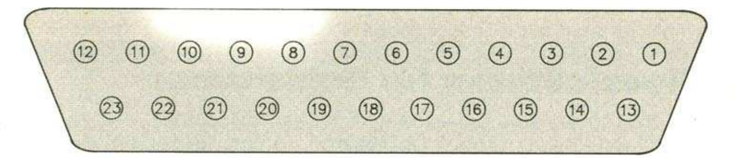
erhalten Sie die Angaben bezüglich des Screens.

(Tobias Helge Kosuch/ub)

Der Schalter für das Laufwerk

Einige Amiga-Programme funktionieren nur, wenn kein externes Laufwerk angeschlossen ist. Aber einige Laufwerke besitzen keinen Ausschalter. Damit Sie sich die Arbeit ersparen, das Laufwerk immer herauszuziehen, bauen Sie sich den Schalter selbst: Sie müssen ihn nur in die RDY-Leitung einlöten — entweder im Laufwerk oder im Anschlußkabel. Das RDY-Signal liegt beim Amiga 500 an Pin 1 des externen Diskanschlusses. Natürlich dürfen Sie den Schalter nur bei ausgeschaltetem Computer betätigen.

(Maurice AL-Khaliedy/ub)



SUB-D-Buchse 23pin/weiblich

Pin	Signal	Beschreibung
1	RDY	Bereitschafts-Signal (active low)
2	DKRD	Schreib/Lese-Signal (active low)
3	GND	Signalmasse
4	GND	Signalmasse
5	GND	Signalmasse
6	GND	Signalmasse
7	GND	Signalmasse
8	MTRXD	Motor-Steuerung (active low)
9	SEL2B	Laufwerk 2 aktivieren (gepuffert, active low)
10	DRESB	Reset-Signal (gepuffert, active low)
11	CHNG	Diskettenwechsel-Signal (active low)
12	+5V	+5 Volt
13	SIDEB	Kopfwahl-Signal (gepuffert, active low)
14	WPRO	Schreibschutz-Signal (active low)
15	TK0	Spur 0-Signal (active low)
16	DKWEB	Schreibfreigabe-Signal (gepuffert, active low)
17	DKWDB	Daten schreiben (gepuffert, active low)
18	STEPB	Signal für Kopf-Stepper (gepuffert, active low)
19	DIRB	Stepper-Richtungs-Signal (gepuffert, high=Richtung Spur 0)
20	NC	-Nicht belegt-
21	SEL1B	Laufwerk 1 aktivieren (gepuffert, active low)
22	INDEX	Index-Impuls (active low)
23	+12V	+12 Volt

Diese Tabelle zeigt die Belegung des 23poligen SUB-D-Steckers für die Zusatzlaufwerke am Amiga

Einschneidende Maßnahmen

CUT/COPY/PASTE benutzt jeder einmal. Der Amiga speichert die Teile des Programms, die mit CUT oder COPY aus einem Listing geschnitten werden, in der RAM-Disk. Der Name der Datei lautet »BasicClip«. Es handelt sich um eine sequentielle Datei. Diese können Sie mit einem Programm lesen und im aktuellen Fenster ausgeben:

```
 DIM w$(1000)
 OPEN "ram:BasicClip" FOR INPUT AS #3
lesen:
 a=a+1
 IF EOF(3) THEN fertig
 INPUT#3,t$
 w$(a) = t$
 GOTO lesen
fertig:
 CLOSE #3
For i= 0 TO a
 PRINT w$(i);
NEXT i
```

Probieren Sie es einmal aus. Sicher finden Sie zahlreiche Anwendungen. Zum Beispiel können Sie ein Unterprogramm schreiben, das den geretteten Block verändert. Danach bauen Sie den angepaßten Block mit PASTE in Ihren Programmen ein. Eine wertvolle Hilfe beim »Debugging«, wenn Sie in einem Block ganz bestimmte Zeichen oder Werte ersetzen wollen. (Birger Fahning/ub)

Shell für alle Fälle

Die Shell des Aztek-C ist eine sehr anwenderfreundliche Sache. Sie erleichtert Programmierern/innen das Arbeiten mit dem CLI und dem Compiler. Wer dieses Utility auch unabhängig vom C-Compiler nutzen möchte, führt folgende Schritte aus:
— Der Shell-Befehl von der Compiler-Diskette wird in den C-Ordner der Arbeits-Diskette kopiert.
— Die Datei »shellseg« aus dem Verzeichnis »L« muß in das gleichnamige Directory auf der Work-Disk.
— Im Verzeichnis »s« benötigt die Shell eine Datei »keys«. Legen Sie die Datei mit dem ED an. »keys« legt die Belegung der Funktionstasten fest. Schreiben Sie beispielsweise folgende Zeilen:

```
KEY1 DIR ^m
KEY2 PATH Ram: ^m
.
.
KEY10 Execute
```

Die Kürzel »^m« bedeuten, daß nach Druck der Funktionstaste der Befehl direkt ausgeführt wird. Im obigen Beispiel entspricht <F1> der Anweisung DIR. Fehlen die Kürzel, so muß noch eine Eingabe von der Tastatur folgen, die mit <Return> abschließt. Die Shell läßt sich nun vom CLI oder über die »Startup-Sequence« aufrufen:

```
SHELL "CON:0/0/600/195/Shell" s/keys
```

Danach arbeiten Sie mit einem CLI, in dem die Funktionstasten belegt sind. Außerdem können Sie alle Anweisungen mit den Cursortasten aus einem Puffer zurückholen, korrigieren und mit <Return> wiederholen. Das spart viel Zeit. (Kilian Frühauf)

Wo stapeln Sie Stapel-Dateien?

Batchfiles sind eine nützliche Sache. Programmierer benutzen sie zum Beispiel, um den Aufruf des Compilers zu vereinfachen. Auch die »Startup-Sequence« ist eine solche »Stapel-Datei«. Durch den CLI-Befehl EXECUTE werden alle Anweisungen in solchen Dateien ausgeführt. Doch wohin mit ihnen? Wenn Programmierer zum Beispiel mit einem Compiler arbeiten, wechseln sie häufig mit CD das aktuelle Verzeichnis und wissen dann nicht mehr, wo eine bestimmte Stapel-Datei steht. Die einfachste Lösung: Schreiben Sie alle häufig verwendeten Batchfiles ins Verzeichnis »s«. Dort sucht der Amiga, wenn er EXECUTE ausführt, immer als erstes.

(Marc Baaden/ub)

»Patchen«

Durch geschicktes Verändern einzelner Bytes läßt sich manches Programm verbessern. Durch »patchen« können Sie ei-

Viele Tips und Tricks widmen sich der Verbesserung eines Programms durch »patchen«: Ein gezielter Patch, und schon öffnet sich das Basic-Ausgabefenster über den gesamten PAL-Bildschirm. Ein weiterer, und schon arbeitet ein Drucker besser. Es gibt viele Anwendungen dieser Technik und verschiedene Arten einen Patch durchzuführen:

Basic-Ausgabe im großen Stil

Das Ausgabefenster des Amiga-Basic hat eine Größe von 200 Punkten in der Vertikalen. Auch mit der Maus läßt es sich nicht auf die volle PAL-Höhe ausdehnen. Der Interpreter ist mit der falschen Maximalhöhe ausgestattet. Doch diese läßt sich durch einen kleinen Patch korrigieren: Stellen Sie zunächst fest, ob Sie die richtige Basic-Version für den folgenden Patch besitzen. Prüfen Sie die Länge des Programms mit Hilfe des CLI-Befehls LIST:

```
LIST AmigaBasic
```

Als Ergebnis sollten Sie 103484 oder 103500 erhalten. In diesem Fall schauen Sie sich das Amiga-Basic genauer an:

```
TYPE AmigaBasic opt h
```

Sobald am linken Bildschirmrand die Adresse $5800 auftaucht, drücken Sie die Leertaste. So halten Sie das Listing an. Mit der BackSpace-Taste setzen Sie die Ausgabe fort. In der Zeile mit der Adresse $57e0 finden Sie den gesuchten Wert für die Fenstergröße:

```
57E0: 000E4E75 2F0A303C 0280323C 00C80C43  ..Nu/.0<
..2<...C
```

Die Parameter $0280 und $00C8 entsprechen der maximalen Breite beziehungsweise Höhe. Diese Werte können Besitzer eines Diskettenmonitors direkt auf der Diskette ändern. Doch nicht jeder hat dieses Werkzeug. Sie können den »Patch« auch mit dem folgenden Basic-Programm durchführen:

```
CLEAR , 50000&
NeueHöhe$ = CHR$(1)+CHR$(0)     'erste 1=256

OPEN "AmigaBasic" FOR INPUT  AS #1 LEN=4096
OPEN "NeuesBasic" FOR OUTPUT AS #2 LEN=4096

PRINT #2, INPUT$(&H57EC,#1);  'Anfang kopieren
PRINT #2, NeueHöhe$;          'neue Höhe
a$=INPUT$(2,#1)          'alte Höhe ignorieren

PRINT #2,INPUT$(&H7000,#1);      'Rest kopieren
PRINT #2,INPUT$(&H7000,#1);
PRINT #2,INPUT$(&H5C00,#1);
WHILE NOT EOF(1)          'spart mühsames Berechnen
 PRINT #2,INPUT$(1,#1); 'der verbleibenden Bytes
WEND

CLOSE #1,#2
```

Wenn Sie Basic vom CLI aus aufgerufen haben, können Sie nach Ablauf des Programms gleich weiterarbeiten. Stellen Sie zunächst sicher, daß das geänderte Basic die gleiche Länge besitzt wie das ursprüngliche:

```
LIST NeuesBasic
```

Bei einem anderen Wert als 103484 beziehungsweise 103500 müssen Sie den »Patcher« überprüfen und von vorne beginnen. Die Funktion des neuen Basic können Sie ausprobieren:

```
RUN NeuesBasic
```

Hat alles geklappt? Wenn alles gut geht, bleiben noch einige Aufräumarbeiten. Sie können mit DELETE die Info-Datei des gepatchten Basic löschen und umbenennen:

```
DELETE NeuesBasic.info
RENAME AmigaBasic AltesBasic
RENAME NeuesBasic AmigaBasic
```

Natürlich sollten Sie diese Arbeiten nicht mit der Original-Extras-Diskette ausführen. Arbeiten Sie mit einer Sicherheitskopie. Dann machen Sie nichts kaputt, wenn Ihnen ein Fehler unterläuft. Es ist zu empfehlen, daß Sie das neue Basic unter dem Originalnamen speichern. Ansonsten müßten Sie die Default-Tool-Namen aller bereits gespeicherten Basic-Programme und den entsprechenden Toolnamen, den Amiga-Basic allen neu gespeicherten Programmen zuordnet, ändern. Doch diese unnötige Arbeit wird sich jeder Programmierer ersparen wollen.

(Peter Pörner/ub)

Anmerkung: Auch vom Amiga-Basic existieren mehrere Fassungen mit anderen File-Längen. Wenn Sie den »Patch« für eine andere Version kennen, schicken Sie uns die Lösung.

Vom Epson-Treiber zum Star NL-10

Wissen Sie, daß Sie die Druckertreiber Ihren Bedürfnissen anpassen können? Sie können einen Druckertreiber patchen. In diesem Beispiel sollten die Besitzer eines Star NL-10 aufhorchen. Der Trick verrät Ihnen, wie Sie einen Epson-Treiber ändern können, damit er Ihren Drucker besser unterstützt. Bisher existiert noch kein spezieller Druckertreiber für den Star NL-10. Doch halb so schlimm, Star-Besitzer können auch den Epson-Treiber verwenden. Aber die Drucker weisen einige Unterschiede in ihren Befehlssätzen auf. So verwendet der Epson für das Ein- und Ausschalten der Fettschrift <ESC,E> und <ESC,F>. Der Star-Drucker arbeitet mit <ESC,G> beziehungsweise <ESC,H>. Diese Sequenzen lassen sich durch einen Patch austauschen:

```
PRINT "Bitte Diskette mit Druckertreiber einlegen"
PRINT "-- arbeiten Sie mit einer Kopie -- "
PRINT "und drücken Sie eine Taste"
WHILE (INKEY$="") : WEND

OPEN "df0:devs/printers/Epson" FOR
INPUT AS #1 LEN=5400
OPEN "df0:devs/printers/Star_NL10Epson" FOR
OUTPUT AS #2 LEN=5400

WHILE NOT EOF(1)
  a$=INPUT$ (1,#1)
  IF a$="E" AND flag=1 THEN a$="G"
  IF a$="F" AND flag=1 THEN a$="H"
  IF ASC(a$)=27 THEN flag=1 ELSE flag=0
  PRINT #2,a$;
WEND
CLOSE #2:CLOSE #1:
PRINT "Fertig !"
```

Mit diesem Verfahren können Sie auch beliebige andere Zeichenfolgen eines Druckertreibers verändern. Probieren Sie es aus, wenn Ihr Drucker nicht so will, wie Sie es möchten.

(Gerhard Miller/ub)

»J« oder »y« so eine Frage

Der neue Befehl des CLI, ASK, ist eine feine Sache. Leider erwartet er ein »y« für die Antwort »Ja«. Da bietet es sich an, den Befehl einzudeutschen. An vier Stellen muß das »y« durch »j« ersetzt werden. Ein Basic-Programm führt alle Änderungen durch:

```
CHDIR "SYS:C"
OPEN "ASK" FOR INPUT AS #8
ASK$ = INPUT$(LOF(8),#8)
CLOSE #8
IF LEN(ASK$) <> 1580 THEN FEHLER
IF MID$(ASK$,1156,1) <> "y" THEN FEHLER
```

mit System!

nen Druckertreiber an Ihren Drucker anpassen oder die Größe eines Fensters (Windows) nach dem Start verändern.

```
IF MID$(ASK$,1180,1) <> "Y" THEN FEHLER
IF MID$(ASK$,1218,1) <> "Y" THEN FEHLER
IF MID$(ASK$,1228,1) <> "y" THEN FEHLER
MID$(ASK$,1156,1) = "j"
MID$(ASK$,1180,1) = "J"
MID$(ASK$,1218,1) = "J"
MID$(ASK$,1228,1) = "j"
OPEN "askd" FOR OUTPUT AS #6
PRINT #6, ask$;
CLOSE #6
END
Fehler:
 PRINT "Da stimmt was nicht"
```

Nach erfolgreicher Konvertierung löschen Sie die Datei »ask.info«. Damit der alte ASK-Befehl erhalten bleibt, heißt der neue Befehl »ASKD«. Wenn Sie ausschließlich mit der deutschen Version arbeiten möchten, löschen Sie ASK, um Speicherplatz zu sparen.

(Harald Zacharias/ub)

Amiga-DOS mit 256 Punkten

Wer mit dem CLI arbeitet, wünscht sich nicht selten, daß das Amiga-DOS-Fenster gleich die volle Bildschirmgröße hat. Dieser Traum kann zumindest für Besitzer des Amiga 1000 wahr werden. Dort befindet sich die Kickstart-Software auf der Diskette und kann mit einem Diskettenmonitor verändert werden. Erstellen Sie zunächst eine Kopie Ihrer Kickstart. Sie müssen drei Blöcke verändern:

```
Block 0454 (Spur 20; Sektor 03; Kopf 1)
1A0: 03434C49 07526573 74617274 0B46696C   .CLI.Restart.Fil
1B0: 6548616E 646C6572 18434F4E 3A302F30   eHandler.CON:0/0
1C0: 2F363430 2F323536 2F416D69 6761444F   /640/256/AmigaDO
1D0: 53000000 012A0000 4E436F70 79726967   S....*..NCopyrig

Block 0455 (Spur 20; Sektor 04; Kopf 1)

000: 20496E63 2E0AA920 31393837 20627920   Inc... 1987 by
010: 41726E65 205A696D 2E210A52 656C6561   Arne Zim.!.Relea
020: 73652031 2E320A00 133A532F 53544152   se 1.2...:S/STAR

Block 0512 (Spur 23; Sektor 06; Kopf 0)
```

Bei der Kickstart-Version 33.166 muß der Block folgendermaßen aussehen:

```
1C0: 000CC0CE 00000B68 000000CD 00000BB0   ...I...h...I....
1D0: 000000CA 00000C7C 000000CB 00000D54   ...E...ö.......T
1E0: 000000D1 00000000 D59B8795 00040000   ...N....O.......
1F0: 00000019 001A001B 001C001D 001E001F   ................
```

Bei der neueren Version 33.180 muß der Block so aussehen:

```
1C0: 000000CE 00000B68 000000CD 00000BB0   ...I...h...I....
1D0: 000000CA 00000C7C 000000CB 00000D54   ...E...ö.......T
1E0: 000000D1 00000000 3304AF78 00040000   ...N....3..x....
1F0: 00000019 001A001B 001C001D 001E001F   ................
```

Es ist wichtig, daß Sie keine zusätzlichen Änderungen vornehmen und nur die gedruckten Werte ändern. Ansonsten stimmt die Prüfsumme in Block 512 nicht mehr. Wenn Sie diesen Wert korrigieren, können Sie natürlich weitere Änderungen durchführen. Beim Zurückschreiben der Sektoren dürfen Sie keine neue Checksumme verwenden, da diese die Kickstartsoftware auf der Diskette zerstören würde.

Wenn Sie mit dieser neuen Kickstart den Amiga füttern und dann eine DOS-Diskette einlegen, meldet sich das CLI mit dem großen Fenster.

(Arne Zimmermann/ub)

Editieren im PAL-Fenster

Große Textfenster erhöhen die Übersichtlichkeit. Ärgerlich, daß das Fenster des Editors ED nicht sofort nach dem Start die volle PAL-Auflösung ausnutzt. Immer wieder müssen die geplagten Anwender die Maus einsetzen. Doch damit soll es ein Ende haben: Mit einem Diskettenmonitor können Sie dem Übel zu Leibe rücken.

Wenn Sie sich die Programmdatei des ED auf der Diskette anschauen, entdecken Sie diese Bytefolge:

```
52 41 57 3A 30 2F 30 2F 36 33 39 2F 31 39 39
```

Der ASCII-Text hierzu lautet:

```
RAW:0/0/639/199
```

Aha, das ist der Schlüssel zum Fenster des Editors. Die Werte in der Befehlsfolge können Sie ändern. Um beim nächsten Aufruf des Editors die volle Bildschirmgröße zu nutzen, müssen Sie die 639 durch 640 beziehungsweise die 199 durch 256 ersetzen. Die letzten 7 Byte lauten:

```
36 34 30 2F 32 35 36
```

Wer diese Änderung auf seiner Diskette vornimmt, kann danach sofort mit dem neuen PAL-ED arbeiten.

(Jürgen Brendel/ub)

Die Lösung mit dem C-Monitor

Wenn Sie das ED-Fenster vergrößern möchten, aber keinen Diskettenmonitor besitzen, nehmen Sie den C-Monitor. Sie können das Fenster zwar auch mit der Maus vergrößern, allerdings verlangsamt sich dann der Bildschirmaufbau des Editors und Sie müssen das Fenster bei jedem Aufruf des ED einstellen. Gehen Sie wie folgt vor: Zunächst starten Sie den Monitor. Geben Sie folgendes ein (nicht die Kommentare und Adressen in Klammern):

```
L c/ED c20000  (absolutes Laden des ED nach $C20000)
e c21aa9       (editieren ab c21aa9. Es erscheint :)
36 <CR>          ('$c21aa9' = 6)
34 <CR>          ('$c21aa9' = 4)
30 <CR>          ('$c21aa9' = 0)
2f <CR>          ('$c21aa9' = /)
32 <CR>          ('$c21aa9' = 2)    ( 35 <CR>)
35 <CR>          ('$c21aa9' = 5)    ( 31 <CR>)
36 <CR>          ('$c21aa9' = 6)    ( 32 <CR>)
   <CR>
S c/ED.256 c20000 c24c6c
```

Nach allen Änderungen öffnet der »ED.256« ein Fenster, das die volle PAL-Auflösung nutzt. Mit der hinter einigen Statements angegebenen Alternative passen Sie den ED an den Interlace-Modus an. Natürlich können Sie auch andere Werte nach eigenem Geschmack verwenden.

(Frank Wille/ub)

So steht es im Lexikon

Patchen: Das nachträgliche Korrigieren von in der Regel Assemblerprogrammen direkt im Maschinencode. Geschieht häufig mit Hilfe eines Debuggers. Das Patchen wird meist nach der Programmfreigabe unter Zeitdruck beim Anwender durchgeführt.

Haben Sie Interesse am »patchen« gefunden? Machen Sie sich selbst auf die Suche. Schauen Sie sich Ihre Programme einmal genau an. Verwenden Sie einen Diskettenmonitor, oder arbeiten Sie mit dem TYPE-Befehl. Schauen Sie sich die Dateien auf der Workbench einmal an. Die Libraries oder die »System-configuration« sind interessante Objekte. Und wenn Sie etwas erkennen, wo es sich eventuell lohnt, ein Byte zu verändern, probieren Sie es aus — natürlich auf einer Kopie des Programms. Falls Sie Erfolg haben und eine nützliche Verbesserung finden, lassen Sie es uns wissen. (ub)

Neue Geheimnisse

Durch Drücken mehrerer Tasten können Sie dem Amiga so manche geheimnisvolle Botschaft entlocken (siehe Ausgabe 2/88, Seite 90).

Eine neue Fingerübung ist die folgende:

Drücken Sie zunächst die linke <Shift>- und <Alt>-Taste. Nicht loslassen. Als nächstes betätigen Sie die <Alt>-, <Shift>- und <Amiga>-Taste auf der rechten Seite. Zum guten Schluß müssen Sie noch eine beliebige sechste Taste drücken; schon versteckt sich die Maus in der rechten unteren Ecke des Bildschirms. Befindet sich die Maus allerdings vor der Fingerakrobatik in der Menüleiste, rutscht der gesamte Workbench-Screen nach unten. Der eigentliche Sinn dieser Kombinationen ist unklar oder kennen Sie eine Anwendung? Was auch immer dahintersteckt, es verbergen sich bestimmt noch mehr solcher geheimen Tastenkombinationen im Amiga. Machen Sie sich auf die Suche.

(Mario Himmel/ub)

ASSIGN — einmal anders

Die Eigenschaften des ASSIGN-Befehls sind allgemein bekannt: Directories und logische Devices lassen sich gezielt bestimmten Verzeichnissen zuordnen. Folgende Besonderheit geht oft unter: Auch Befehle lassen sich einem Namen zuordnen.

```
ASSIGN X: C:EXECUTE
```

Um die »Startup-Sequence« ausführen zu lassen, genügt:

```
X: startup-sequence
```

Ähnlich lassen sich alle anderen CLI-Statements behandeln. Dieser Weg ist dem Umbenennen von Befehlen mit RENAME vorzuziehen. Die alten Namen bleiben nämlich zusätzlich erhalten.

(Marc Baaden/ub)

Ein interessantes Projekt

Es gibt unterschiedliche Arten von Piktogrammen, unter anderem die »Tools« und die »Projects«. Tools sind Programme. Durch zweimaliges Anklicken werden sie gestartet. Zwei Beispiele hierfür sind die »Preferences« und die Uhr.

Projects dagegen sind Dateien, die von anderen Programmen erzeugt wurden. Sie können nicht eigenständig geladen werden. Sie sind abhängig von einem Tool. Jedes Basic-Programm stellt ein solches Projekt dar. Dennoch lassen sich die Projekte durch zweimaliges Anklicken aktivieren. In der Info-Datei des Piktogramms ist der Name des zugehörigen Tools vermerkt. Sie verändern ihn mit Hilfe des Menüpunktes »Info« auf der Workbench:

Angenommen, Sie haben mit einem Editor einen Text verfaßt und möchten, daß dieser beim Anklicken des Datei-Piktogramms auf dem Bildschirm erscheint. Kopieren Sie zunächst das Tool »More« von der ExtrasD-Diskette in die RAM-Disk. Als Mutterprogramm tragen Sie »ram:more« in der Info-Datei Ihres Textes ein. Nun müssen Sie nur noch das Text-Icon anklicken.

Viele Editoren erzeugen zu den gespeicherten Dateien allerdings keine Piktogramme. Ein Beispiel ist »MicroEmacs« auf der Extras-Diskette. Doch der Workbench-Benutzer kann zu diesen Texten ein bereits existierendes Icon verwenden. Er darf sogar ein Icon wählen, dessen Gattung nicht dem geforderten Projekt-Typ entspricht:

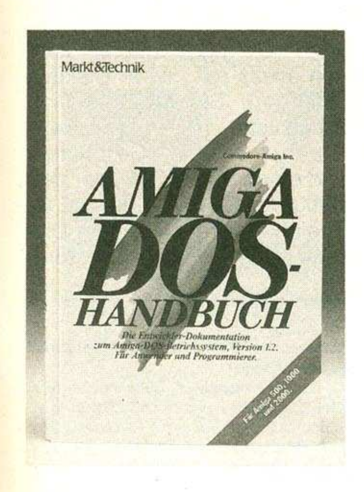
Laden Sie in diesem Fall »IconEd« sowie das gewünschte Bild. Klicken Sie einen zweiten Rahmen an, in den Sie ein beliebiges Piktogramm laden, dessen Typ aber dem gewünschten entspricht — in diesem Fall ein Projekt. Löschen Sie den zweiten Kasten mit »Clear Image«, wählen Sie »Merge with Frame« und bewegen die Maus auf das erste Feld. Wenn Sie nun die Maus loslassen, befindet sich im aktivierten zweiten Quadrat das gewünschte Image. Dieses speichern Sie unter dem Namen des Textes als Info-Datei.

(Ralph Benzinger/ub)

Amiga-DOS-Handbuch

Dieses Buch beinhaltet mit dem DOS-Handbuch, dem DOS-Programmier-Handbuch und dem technischen DOS-Handbuch drei aus dem englischen übersetzte Werke. Bereits einmal ist diese Systemdokumentation, allerdings die englischen Fassungen, zusammengefaßt worden (siehe Amiga-DOS-Manual, Amiga-Magazin 11/87).

Das erste Buch, das Anwender-Handbuch des Amiga-DOS, erklärt ausführlich den grundsätzlichen Umgang mit den CLI-Befehlen. Dieser Buchteil ist in zwei große Abschnitte gegliedert. Allgemeine Handhabung, Datenstrukturen und andere Grundlagen-Informationen zum Amiga-Betriebssystem werden im ersten Teil mit vielen gut verständlichen Beispielen erläutert. Der zweite Teil ist eine alphabetische, tabellarische Auflistung aller DOS-Befehle. Dieser Teil ist wie ein Nachschlagewerk gestaltet. Jedem Befehl ist eine extra Seite gewidmet. Selbst-



verständlich handelt es sich hier um die Befehle der Betriebssystemversion 1.2. Unverständlich ist die verhältnismäßig große Abhandlung über den Zeileneditor »EDIT«. Selbst der Bildschirmeditor »ED«, der den »EDIT« inzwischen vollständig ersetzt hat, wird immer weniger eingesetzt. Das erste Buch ist damit für den CLI-Einsteiger bestens geeignet, obwohl einige erklärende Beispiele wünschenswert gewesen wären.

Das DOS-Programmierer-Handbuch erleichtert dem schon fortgeschrittenen DOS-Anwender die Systemprogrammierung. Neben der Programmierung unter Amiga-DOS findet der Leser dort auch Informationen über den Einsatz einer SUN-Workstation oder eines IBM-PC zur Programmentwicklung. Der Umgang mit den DOS-Routinen, das Ausführen von Programmen, der Makro-Assembler, C und die Ein-/Ausgabe über ein Terminal sind weitere Themen. Die Autoren handeln die einzelnen Themen sehr kompakt ab. Erläuternde Beispiele sind leider selten.

Der dritte Teil dieses Werkes, das technische Handbuch, ist nur noch für den Programmierer interessant. Es enthält Informationen über Aufbau und Struktur des DOS-File-System sowie Hinweise für den fortgeschrittenen Entwickler. Die Angaben zum File-System umfassen Informationen über Rootblock, User Directory Block, File Header Block, File List Block und Data Block. Ein weiteres Thema ist die Strukturbeschreibung von Objektfiles, wie sie von Assembler und Compilern erzeugt werden, sowie den Loadfiles.

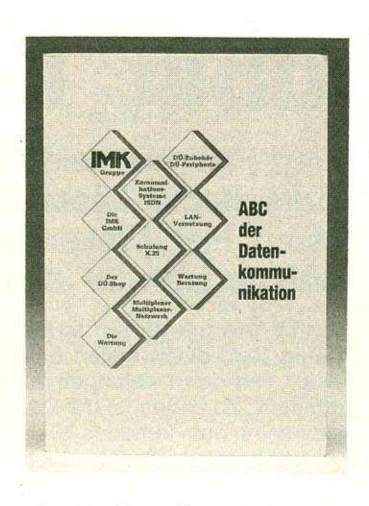
Das Amiga-DOS-Handbuch bietet viel Informationen in straffer und gut strukturierter Form. Die Zusammenfassung von Informationen für Einsteiger als auch angehende Profis hat Vor- und Nachteile. Es ist zwar für viele Leser etwas dabei, der Profi muß allerdings längst bekanntes Wissen dazukaufen. (Joschy Polierer/pa)

Commodore-Amiga Inc, Amiga-DOS-Handbuch, Markt & Technik Verlag AG, 342 Seiten, ISBN 3-89090-465-3, Preis 59 Mark

Das ABC der Datenübertragung

Hätten Sie gedacht, daß sich mit den Fachwörtern und Abkürzungen der Datenfernübertragung ein Buch mit 176 Seiten füllen läßt? Eine Ahnung davon können Sie bekommen, wenn Sie sich als Gast in einer Mail- beziehungsweise Chatbox aufhalten, in der andere Teilnehmer mit Wörtern um sich werfen, von denen Sie selbst mit überdurchschnittlichem Computerwissen noch nie etwas gehört haben. Das »ABC der Datenübertragung« hilft Ihnen, sich bei einer derartigen Kommunikation zu Wort zu melden, ohne gleich als Anfänger erkannt zu werden.

Das Buch ist in drei Teile gegliedert. Diese sind durch verschiedene Einfärbung der Seiten leicht zu unterscheiden. Auf den »gelben Seiten« des ersten Teils klären die Autoren über die Bedeutung selbst we-



niger gebrauchter Abkürzungen auf. Der Leser findet so ungewöhnliche Kürzel wie AYT (Are you there? — Bist du hier?). Die weißen Seiten des zweiten Teiles enthalten Grundbegriffe der Datenfernübertragung. Die Erklärungen sind trotz knapper Formulierung gut verständlich. Selbst der fortgeschrittene Anwender kann die Erläuterungen zu einem so schwierigen Thema wie dem Protokoll x.25 nutzen. Mit den blauen Seiten des Wörterbuchs im letzten Teil erleichtern die Autoren den weniger englischsprechenden Computerfans den Umgang mit Fachbegriffen. Dieser Teil enthält die deutsche Übersetzung für wichtige Stichworte aus dem Sachgebiet der Datenfernübertragung.

Das Buch kann sowohl dem angehenden als auch praktizierenden »Mailbox-Crack« empfohlen werden. Im handlichen Taschenformat leistet es als Nachschlagewerk in unmittelbarer Nähe des »Terminals« wertvolle Dienste.

(R. Seitz/pa)

Das ABC der Datenkommunikation, DATA-COM Buchverlag, ISBN 3-892-38-008-2, Preis 38,50 Mark

Das große Buch zu Amiga-DOS

Dieses Werk bietet eine ausführliche Einführung in den Umgang mit dem CLI, der zweiten Benutzerschnittstelle des Amiga, sowie Informationen für den fortgeschrittenen DOS-Anwender. In der Einführung des in acht Teile gegliederten Buches gehen die Autoren auf die Aufgabe des Amiga-DOS, das Öffnen eines CLI-Fensters und die Help-Funktionen ein. Der zweite Teil befaßt sich mit den einzelnen CLI-Kommandos. Hinweise zum täglichen Umgang mit dem CLI bietet der dritte Teil.

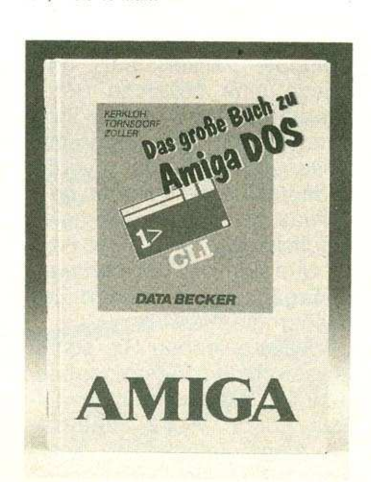
Die Teile 4 bis 6 behandeln die Themen Batchprocessing, Multitasking und interne DOS-Bibliotheken. Eine Reihe neuer DOS-Befehle zum Abtippen befindet sich in Form von DATA-Listings im siebten Abschnitt. Teil 8 schließlich enthält eine Kurzerklärung der DOS- und Editor-Kommandos zum Nachschlagen.

Die Autoren erklären die einzelnen Befehle anhand vieler Beispiele. Hier unterscheidet sich das Amiga-DOS-Buch von anderen Büchern zum selben Thema. Die Befehle werden ausführlich erklärt und nicht in der sonst üblichen, tabellarischen Form beschrieben. Dies ist für den Leser sehr angenehm. Er kann dieses Buch dadurch wirklich »lesen« und nicht nur als reines Nachschlagewerk verwenden.

Im von vielen Autoren des Becker-Verlages gewohnten, lässig saloppen Schreibstil springen die Autoren von Thema zu Thema. Der Einsteiger läuft Gefahr, einen aufgenommenen Faden zu verlieren. Ungünstig gewählte Beispiele tragen dazu bei: So wird die Funktion der Unterverzeichnisse und Pfade am Beispiel einer Festplatte erklärt, die wohl kaum im Besitz der meisten Leser sein dürfte. Viele der angeschnittenen Probleme und Themen können von den Autoren selbst nicht ausreichend erklärt werden oder sind nur unzureichend erprobt. Beim Lesen der Kapitel Multitasking und Multiuserbetrieb kann man sich manchmal des Eindrucks nicht erwehren, daß es sich hierbei um noch unerforschte Gebiete handelt. Dennoch bietet das Buch für experimentierfreudige Computerfans eine Reihe interessanter Anregungen.

(Joschy Polierer/pa)

Das große Buch zu Amiga-DOS, Kerkloh/Tornsdorf/Zoller, Data Becker GmbH, 310 Seiten, Preis 49 Mark



Die Public Domain-Seite

Fred Fish hat seine Public Domain-Serie aufgestockt. Lesen Sie, was auf den neuesten Fish-Disketten zu finden ist und welche sich für Ihre Sammlung eignen.

Der letzte Stand: Die Fish-Reihe umfaßt nun 127 Disketten. Allerdings enthalten die letzten 15 bis 20 Disketten vorwiegend Animationsdemos und Demodateien, die aber gut genug sind, um sich näher damit zu befassen.

Ein Filemonitor, mit dessen Hilfe Dateien geladen, editiert und wieder gespeichert werden, findet sich auf »AUGE 4000 Nr.11«. Der »FileZap« ist zwar nicht sonderlich leistungsstark, genügt aber vollauf, wenn an einer Datei kleine Manipulationen zu tätigen sind. Außerdem bekommt der Käufer dieser Diskette verschiedene IFF-Bilder, hilfreiche Tools (Hilfsprogramme) und das Spiel »WBLander«, das einfach aufgebaut ist.

Hauptsächlich an Musikliebhaber richtet sich die »TBAG 15«, die einen Sonix-Player sowie einige damit abspielbare Musikstücke enthält. Zwar sind auf der Diskette noch weitere, verschiedene Programme enthalten, doch ist diese Diskette wohl nur der Sounds wegen lohnenswert.

Für die Grafiksammlung

Viele schöne, detailliert gezeichnete Bilder im IFF-Format, zu einer Diashow zusammengefaßt, bekommt man beim Kauf der »Tornados 27«. Wenn daran gedacht wird, sich eine Grafik-Bibliothek aufzubauen, sollten diese Bilder berücksichtigt werden.

Wer kennt nicht das äußerst hilfreiche Grafik-Hilfsprogramm »Butcher«, mit dessen Hilfe sich Bilder vielfältig nachbearbeiten lassen. Zwar kein Ersatz dafür, doch ebenfalls sehr vielfältig in seinen Funktionen ist der »IFF-Image-Filter« (Tornados 25), der wohl für jeden Computergrafiker auf dem Amiga eine nützliche Bereicherung seiner Grafik-Tools darstellt. Mit dem Programm lassen sich Farben verfälschen, Bilder verfremden oder verwaschen. Durch die vielen Möglichkeiten erhält der Käufer eine sinnvolle und preiswerte Ergänzung zu seinen Zeichenprogrammen.

Vielleicht haben Sie schon etwas von dem Vorspanngenerator für Videovorspänne »TV-Text« gehört. Wer sich für das Programm interessiert, sollte sich zuerst die Demoversion von TV-Text zulegen, um so mit wenig Geld einen ersten Eindruck von den Leistungen zu erhalten. Zu finden ist die Demoversion auf »Tornados 10«.

Musikfreaks werden auch mit der »AUGE 4000 Nr.13« etwas anfangen können: Sie enthält drei verschiedene, digitalisierte Musikstücke und eine sehr hübsch gemachte Grafikanimation. Diese Diskette ist sicher nicht das Gelbe vom Ei, ist aber als Demonstrationsdiskette gut geeignet.

Kennen Sie die Simulation »Life«, bei der nach festgelegten Regeln Entwicklungsvorgänge simuliert werden? Fish »111« enthält eine neue Life-Simulation, die den gerade bestehenden Workbench-Screen als Ausgangsgrundlage, also als Startgeneration hernimmt. Weiter enthält diese Diskette einen Mandelbrot-Generator und weitere, mehr oder weniger nützliche Routinen.

Vorwiegend Spiele erhält der Käufer der Fish »120«. Beispielsweise eine »Space Invaders«-Version oder ein spielstarkes Backgammon. Es besitzt eine gute Grafik und macht sicher sehr viel Spaß, da es nicht so ohne weiteres zu besiegen ist. Auch ein Strategiespiel ist enthalten. Als kleine Draufgabe bekommt man noch eine Routine, mit der Icons manipuliert werden können. Die Diskette ist sicher kein Muß, kann aber des Backgammons wegen gekauft werden.

Für Basic-Programmierer ist Fish »121« anzuraten. Die Diskette enthält verschiedene Basic-Routinen, unter anderem einen Funktionenplotter und ein Programm, um Basic-Programme von anderen Computern korrekt auf den Amiga zu übertragen. Außerdem findet sich auf dieser Diskette die neueste Version des Editors »UEdit«, der eine große Auswahl an Funktionen bietet.

Haben Sie sich das auch schon immer gewünscht: ein Spiel, das IFF-Bilder lädt und diese dann in ein Puzzlespiel umwandelt? Dann sollten Sie sich die Fish »122« zulegen. »IFF2PCS« tut genau das eben Beschriebene. Es lädt IFF-LoRes-Bilder und zerlegt diese in lauter Einzelteile, wobei die Größe und Anzahl der Puzzleteile bestimmt werden kann. Danach dürfen Sie mausgesteuert das Bild wieder zusammensetzen. Wir meinen, daß das Programm so reizvoll ist, daß es in keiner Spielesammlung fehlen sollte.

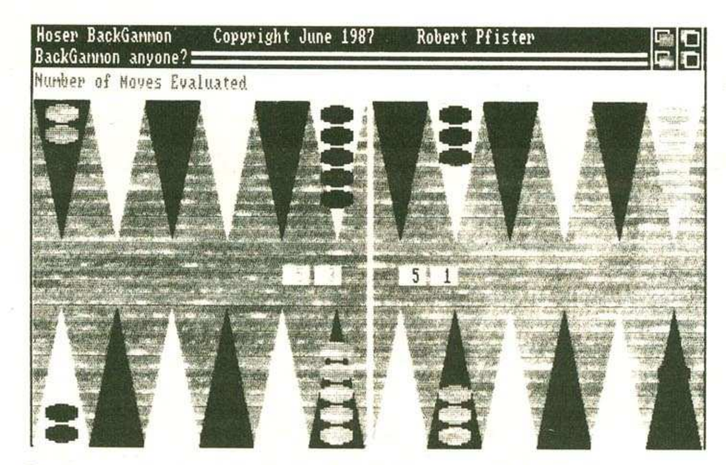
Das CLI wird schneller

Ein äußerst sehenswertes Videoscape-Animationsdemo findet sich auf Fish »123«. Es zeigt wieder einmal eindrucksvoll die Grafikleistungen des Amiga. Außerdem enthält die Diskette Ersatz für viele CLI-Befehle. Die neuen Kommandos machen alles, was auch die alten Befehle tun, nur schneller. Des Komforts wegen, den diese »Replacements« bieten, sollte diese Diskette auf jeden Fall gekauft werden.

Legen Sie gerne Karten? Dann sollten Sie sich einmal das Amiga-Basic-Spiel »Tarot« auf Fish »124« ansehen. Das Spiel kann sicher in die eine oder andere Spielesammlung mit aufgenommen werden. Des weiteren finden sich noch viele Icons (Bildsymbole) auf dieser Diskette, die man übernehmen und für eigene Programme verwenden kann.

Grafikfreunde kommen bei Fish »125« auf ihre Kosten: Das bekannte Animationsdemo »ElGato«, das eine laufende Katze zeigt, bei der sich zusätzlich die Projektionsfläche um die eigene Achse dreht, sorgt sicher für einige verwunderte Ausrufe über die Leistungsfähigkeit dieses Grafikcomputers. Das Demo, das immer wieder für Erstaunen sorgt, findet Platz in jeder PD-Sammlung.

Auch Fish »127« ist für Grafikfreaks interessant: Gleich mehrere Animationsdemos sind darauf zu finden. Aber auch C-Programmierer können etwas mit dieser Diskette anfangen. Sie erhalten eine zusätzliche, frei verwendbare Library, diverse C-Routinen und einen Virus-Killer, der den lästigen SCA-Virus sowohl im Speicher als auch auf der Diskette erkennt. (dm)



Backgammon — ein Spiel für Strategen (Fish 120)

Adressen der PD-Anbieter (ohne Anspruch auf Vollständigkeit):
AB-Computersysteme, A. Büdenbender, 5000 Köln 41, Wildenburgstr. 21, Tel. 0221/4301442
EcoSoft AG, 7890 Waldshut, Kaiserstr. 21, Tel. 07751/7920
Stefan Ossowski, Veronikastr. 33, 4300 Essen 1, Tel. 0201/788778
Datentechnik Bittendorf, 6360 Friedberg, Postfach 100248, Tel. 06031/61950
M.A.R., Peter Rauscher's Computershop, Weldengasse 41, A-1100 Wien, Tel. 0222/621535 (aus Deutschland bitte Vorwahl von Österreich vorwählen)
B.K.D., Bernd Küpper, Felberstr. 7, A-5730 Mittersill, Tel. 06562/282
Fastworks, 5090 Leverkusen, Fichtestr. 16, Tel. 0214/94668
Markus Scheer, 4630 Bochum 5, Kapellenweg 42, Tel. 02934/411958
Uwe Schmielewski, 4100 Duisburg, Haroldstr. 71, Tel. 0203/376448
PD-Shop, 4018 Langenfeld, Opladener Str. 30
Atlantis, 5030 Hürth 8, Dunantstr. 53, Tel. 02233/31066
Rainer Wolf, 4420 Coesfeld, Deipe Stegge 187, Tel. 02541/2874
Soyka Datentechnik, 4630 Bochum 5, Hattinger Str. 685, Tel. 0234/411913
Christian Bellingrath, 5860 Iserlohn, Trift 10, Tel. 02371/24192
Intersoft, 4200 Oberhausen 1, Nohlstr. 76, Tel. 0208/809014
A. Fischer, 4794 Hövelhof, Kirchstr. 40, Tel. 05257/4347
AUGE 4000, c/o U. Trempelmann, Lochnerstr. 24, 4030 Ratingen, Tel. 02102/23371

Nimm

AMIGA test

Modula-2 ringt mit Basic, C Amiga-Programmierer. Nun der Programmiersprachen zu

Gute 20000 Zeilen Quelltext pro Minute verarbeitet »M2Amiga«. Der schnelle Modula-2-Compiler ist die ausgewachsene Fassung des Public Domain-Compilers, der die Grundlage zum Modula-2-Kurs in der AMIGA bildet (siehe Seite 98).

Mit dem Öffnen des Umschlags, in dem sich die Diskette zu M2Amiga befindet, sind Sie bereits in den Club der M2Amiga-User aufgenommen. Erst später werden Sie im deutschen Handbuch lesen, daß Sie durch das Brechen des Siegels die Lizenzbedingungen von A.+L.Meier-Vogt bereits anerkannt haben.

Erfreulich ist, daß sich die Diskette über die Workbench oder das CLI kopieren läßt. Sie können jederzeit eine Sicherheitskopie machen. Das Original enthält über 800 KByte an Software:
— ein Installationsprogramm,
— den Editor »m2emacs«,
— den Compiler,
— den Linker,
— eine umfangreiche Bibliothek sowie
— instruktive Beispiele als Quelltext.

Mach mich

»M2Amiga« erweist sich auf allen Amiga als lauffähig, vorausgesetzt, es stehen mindestens 512 KByte an RAM bereit. Sie sollten auch zwei Laufwerke besitzen. Eine vollständig vorbereitete »M2Amiga-Diskette« ist nämlich bereits zu 90 Prozent gefüllt. Bevor der Modula-2-Willige das erste Programm erstellen kann, muß er das System konfigurieren. Dazu formatiert er eine Diskette, bootet mit der mitgelieferten Diskette, ruft das Programm »DoMe« von der Workbench aus auf und folgt dessen Anweisungen. 15 Minuten benötigt »DoMe«, um eine Systemdiskette anzulegen, die nach einer Behandlung mit INSTALL bootfähig wird. »DoMe« ist aber auch das einzige »M2Amiga«-Programm, das im Schneckentempo abläuft.

Damit ist das System aber noch nicht vollständig konfiguriert. Das Handbuch sollte nicht mit weiterer Anleitung geizen. Es sind unter anderem noch die Dateien »S/StartUp-Sequence« und »S/M2Path« zu vervollständigen, das Unterverzeichnis »C« um Hilfsprogramme zu ergänzen, das Unterverzeichnis »Devs/Printers« anzulegen und der Treiber für den Lieblingsdrucker zu kopieren.

In der »Startup-Sequence« sollten Sie die Größe des Stack auf 20 KByte festlegen und mit ADDBUFFERS die Puffer für den Datenverkehr mit den externen Speichern auf mindestens 32 KByte erhöhen. Damit kommen Sie der hohen Arbeitsgeschwindigkeit von Compiler und Linker entgegen.

Das gesamte Entwicklungssystem von M2Amiga läßt sich sowohl von der Workbench als auch vom CLI einsetzen:
— Die Workbench-Ebene bietet ein beispielhaft hohes Maß an Integration der Programmierwerkzeuge und gestattet deren leichte Bedienung.
— Alle Optionen von Compiler, Linker und Betriebssystem können jedoch nur von der CLI-Ebene aus genutzt werden.

»Speicherkönigen« mit mehr als 512 KByte RAM empfehlen wir, alle zur Programmentwicklung erforderlichen Programme in der RAM-Disk — am besten resetfest — zu speichern. Auch der Editor »m2emacs« arbeitet unter diesen Bedingungen im Gegensatz zu TDI-Modula einwandfrei. Eine Beschreibung des Editors (und auch der Sprache Modula-2) finden Sie im Modula-2-Kurs auf Seite 98. Hier wollen wir nur einige Besonderheiten von m2emacs erwähnen:
— Er arbeitet in einem eigenen Screen und nutzt den Bildschirm voll aus.
— m2emacs reserviert sich Speicherplatz nach Bedarf. Die Länge der bearbeitbaren Dateien wird nur durch den verfügbaren Speicher begrenzt.
— Der Löschpuffer paßt sich der Länge des zu sichernden Textes an.
— Sie können mehrere Dateien gleichzeitig in separaten Puffern halten. Der »Screen« ist teilbar, um zwei Texte gleichzeitig anzuzeigen.
— Selbst bei der Bearbeitung von umfangreichen Dateien (>70MByte) arbeitet der Editor zügig.
— Fehler beim Compilieren werden im Quelltext positionsgerecht mit einer deutschen Fehlermeldung angezeigt. Das System hält etwa 500 Fehlermeldungen bereit. Wenn Sie allerdings Ihren Lieblingseditor einsetzen, lassen Sie die Fehlerliste mit dem Hilfsprogramm »m2error« anzeigen.

M2emacs ist ein praktisches Werkzeug. Modula-2-Programmierer vermissen höchstens eine Funktion, um die Groß- beziehungsweise Kleinschreibung einzelner Buchstaben elegant zu wechseln. Leider operiert m2emacs auch nur mit einem Makro und die Dokumentation ist etwas dürftig.

Wertvolles Insider-Wissen

Der Compiler »m2c« erzeugt aus dem Modula-2-Quelltext den Objektcode. Bevor wir in die Tiefe des »m2c« vordringen, sei festgestellt, daß er präzise, zuverlässsig und schnell arbeitet — ein ideales Instrument, um anspruchsvolle Software zu erstellen. Die Handhabung ist ausreichend im Handbuch beschrieben.

»m2c« meckert zwar häufiger als dies bei vergleichbaren Produkten der Fall ist, dafür erhalten Sie recht sicheren Code. Bei der Portierung einiger Programme spürt »m2c« Fehler auf, die der Modula-2-Compiler von TDI toleriert.

Der Compiler unterstützt den vollen Sprachumfang von Modula-2. Sich abzeichnende Änderungen, des noch nicht endgültig festgeschriebenen Standards, wurden vorab realisiert. Das Entwicklungsteam hat »Insiderkenntnisse«, es arbeitet in der entsprechenden Standardisierungsgruppe an der ETH-Zürich aktiv mit. Durch den im Handbuch zugesagten »Update-Service« scheint gewährleistet zu sein, daß die M2Amiga-User von künftigen Entwicklungen partizipieren werden:

Beispielsweise sind künftig keine Prioritäten für Module mehr vorgesehen. Aus Gründen der Portabilität können Sie zwar Prioritäten in der Kopfzeile eines Moduls angeben, die aber nicht berücksichtigt werden. Bei Bedarf setzen Sie die Priorität mit Hilfe der Prozedur »Exec.SetSR«.

Die Aufstellung der reservierten Worte enthält unter anderem CODE, FORWARD und REM. Sie gehören nicht zum Standard. Es handelt sich um implementationsbedingte Erweiterungen.

Zu den vordefinierten Bezeichnern ist anzumerken, daß der Typ »BOOLEAN« intern die Werte »-1« (TRUE) und »0« (FALSE) annehmen kann. Dies erleichtert den Einsatz im Zusammenhang mit Routinen des Betriebssystems. Nach außen sollte »BOOLEAN« wie gewohnt erscheinen, was zum Zeitpunkt des Testes jedoch nicht der Fall war. Außerdem ist »NIL« als »0« definiert (häufig üblich »-1«).

Auch die verwendeten Datentypen entsprechen dem zu erwartenden Standard. M2Amiga arbeitet mit folgenden Typen: CARDINAL, INTEGER, LONGCARD, LONGINT, REAL (IEEE), LONGREAL (IEEE). Zusätzlich ist im Modul »System« der Typ FFP definiert (Fast Floating Point Standard von Motorola).

Typisch Modula-2

Wichtig in diesem Zusammenhang ist das Thema Typenumwandlung: Compilerbauer unterscheiden zwischen Typenkonversion und Typentransfer. Unter Konversion verstehen sie den Wechsel des Typs einer Variablen unter Beibehaltung der »Bitbreite«, der Bereichsgrenzen und der inneren Darstellung. Ein Beispiel:

```
VAR C : CARDINAL;
(* 1 Wort breit *)
    B : BITSET;
(* 1 Wort breit *)
    C := CARDINAL(B);
VAR L : LONGINT;
    R : REAL;
    L := 5;
    R := REAL(L);
(* R enthält 5.0 *)
```

M2. . .

und Assembler um die Gunst der setzt »M2Amiga« in diesem Kampf einem entscheidenden Wurf an.

Die Restriktion von Konversionen sollen durch den (unsauberen) Typentransfer überwunden werden. Dazu dient die Prozedur »SYSTEM.CAST«, die voraussichtlich in den Modulastandard eingehen wird: »CAST« soll es somit ermöglichen, zwischen Typen unterschiedlicher Breite hin- und herzuschalten. Leider ließen sich die dabei zu erwartenden Nebenwirkungen nicht untersuchen, da die getestete Version noch voraussetzte, daß Ursprungs- und Zieltyp den gleichen Speicherplatz einnehmen.

Der Sprachumfang von M2Amiga wird durch hardwarespezifische Elemente ergänzt, die im Pseudomodul »SYSTEM« beziehungsweise »Coroutines« angelegt sind. »SYSTEM« ist ausdrücklich zu importieren, um auf die Abhängigkeit von der Hardware aufmerksam zu machen. Die Vorschläge von Wirth [1] wurden — mit einer Ausnahme — realisiert:

— Die Prozedur »IOTRANSFER« ist nicht mehr vorhanden.

— Überraschend aber beabsichtigt ist, daß die Standardprozedur »ADR« neben Variablen auch auf Konstanten anwendbar ist.

— »SYSTEM« enthält den Datentyp »ADDRESS«, der erfreulicherweise als »POINTER TO BYTE« deklariert ist und damit der Programmierung des Betriebssystems entgegenkommt.

— Das Einfügen von Maschinencode in Quelltexte ermöglicht die Prozedur »INLINE«.

Des weiteren stehen Prozeduren zur Verfügung, um Register zu laden (»SETREG«) oder ihren Inhalt auszugeben (»REG«). Auch eine »SHIFT-Prozedur« ist anzutreffen, mit der Sie Werte bitweise schieben können. Sie sind also ohne weiteres in der Lage, die Register eines Uhrenchips anzusteuern oder einen Treiber für Diskettenlaufwerke zu entwickeln. Listing 1 enthält ein Beispiel für das Anlegen von Tabellen, in dem maschinennah gearbeitet wird.

Stark rekordverdächtig

Die im Vorspann gemachte Angabe stimmt tatsächlich: M2Amiga bearbeitet rund 20000 Zeilen Quelltext pro Minute, nachdem die Importe geladen sind. Die hohe Geschwindigkeit resultiert aus dem »Single-Pass-Konzept«. Schnelle Compiler verlangen aber auch ihren Preis. Alle Objekte müssen vor der ersten Verwendung bekannt sein. Eine einzige Ausnahme besteht für die Deklaration von »POINTER-Typen«. Für die vorläufige Deklaration später folgender Prozeduren verwenden Sie das von Pascal her bekannte reservierte Wort »FORWARD«.

In der Theorie erzeugen Compiler mit mehreren Durchläufen dichteren und schnelleren Code, da die Gesamtstruktur eines Programms vor der Codegenerierung bekannt ist. In Listing 2 zeigen wir an einem einfachen Beispiel, wie sich die Technik eines »Single-Pass-Compilers« nachteilig auf den erzeugten Code auswirken kann.

In dem Beispiel ist eine einfache FOR-Schleife realisiert. Eine Variable soll 10000mal mit 3 multipliziert werden. »Aztek-C« erledigt die Aufgabe mit 28, »TDI« mit 36 und »M2Amiga« mit 32 Byte. Es werden für einen Schleifendurchlauf 134 (»Aztek-C«), 160 (»TDI«) und 138 (»M2Amiga«) Taktzyklen benötigt. Daraus folgt, daß in diesem Beispiel der Code des »m2c« um eine halbe Mikrosekunde langsamer ist, als der vom »Aztek-C-Compiler«. Aufgrund der günstigeren Wahl der Adressierungsarten ist M2Amiga aber schneller als TDI.

Eine der Restriktionen des »m2c«, die übrigens beispielhaft dokumentiert sind, erweist sich eher als Stärke: Die Länge eines Moduls ist auf 32 KByte begrenzt. Dies ermöglicht die Anwendung zeitlich vorteilhafter Adressierungsarten. Da Programme aus beliebig vielen Modulen zusammengesetzt sein dürfen, wirkt sich die Einschränkung praktisch kaum aus. Daraus folgt, daß die Vorteile, die C-Programme für die Adressierung aus dem kleinen Speichermodell ziehen, für »m2c-Programme« stets gegeben sind.

Der Compiler wartet aber noch mit einer weiteren Überraschung auf: Die Entwickler haben sich der Assemblerprogrammierung erinnert. Erstmalig wurde eine zeitsparende Möglichkeit für den Amiga realisiert, Parameter an Prozeduren in Registern zu übergeben und nicht den Stack einzusetzen. Mit dieser Technik kann viel Zeit eingespart werden. Ein einfaches Beispiel für das Register »D2«:

```
VAR Li : LONGINT;
PROCEDURE P1(d{2} :
CARDINAL) : LONGINT;
BEGIN
   RETURN REG(2);
END P1;
   Li := P1(23H);
```

Für die Rückgabe des Funktionswertes an die Variable »Li« wird automatisch das Register »D0« verwendet, entsprechend der Konventionen von »C-Compilern« für den Amiga. Innerhalb der Prozedur »P1« erfolgt somit die Übergabe des Registerinhaltes von D2 nach D0. Leider ist das parameterübernehmende Register nicht als Registervariable im Sinne von C einsetzbar. Es wird auch empfohlen, Register nicht als Parameter in normalen Prozeduren zu verwenden.

Die Väter von »m2c« hatten ebenfalls den bewundernswerten Gedanken, Sprünge zu Systemroutinen nicht wie übliche Prozeduraufrufe zu realisieren. Sie erkannten, daß es unter Benutzung der im vorigen Absatz beschriebenen Technik nicht mehr erforderlich ist, für den Aufruf von Systemroutinen Implementationsmodule anzulegen. Voraussetzung ist: Der erforderliche Code muß nach dem Import von Definitionsmodulen erzeugt werden können. Die Pseudoprozedur »SYSTEM.CODE« bewirkt einen Sprung zu einer Subroutine (indirekte Adressierung mit Distanz). Dazu wird das Register A6 eingesetzt. Listing 4 stellt den Sachverhalt in einem Beispiel dar.

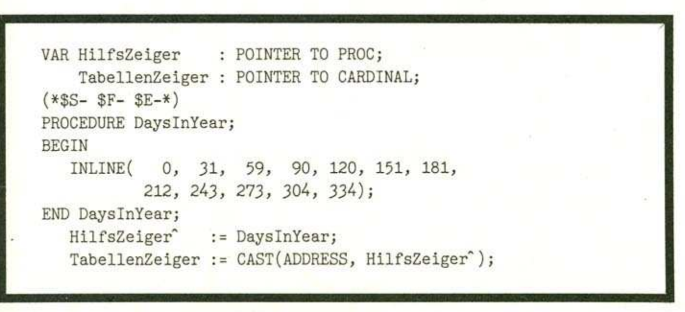
Der Inhalt des Registers D0 würde in dem Beispiel nach Rückkehr aus der Subroutine als Resultat verwendet.

Geistesblitze

Weiterhin wichen die Entwickler von der formalen Forderung ab, daß zu jedem Definitionsmodul eine entsprechende Implementation vorhanden sein muß. Für externe Module, die nur Definitionen und keine Prozedurköpfe enthalten (Verfahren "CODE" entsprechend des vorherigen Absatzes ausgenommen), ist es nicht erforderlich, eine Implementation anzulegen. Ein Implementationsmodul wäre in diesen Fällen ohnehin »leer«. Dazu ist eine bestimmte Compileroption zu setzen (*$M-*), die »m2c« instruiert so zu arbeiten, daß der Linker nichts vermißt. Die Vorteile dieses Vorgehens liegen auf der Hand:

— sämtlicher Code, der erforderlich ist, um Systemroutinen aufzurufen, wird bereits während des Compilierens erzeugt sowie in das Compilat eingefügt,

— der so erstellte Code ist



```
VAR HilfsZeiger    : POINTER TO PROC;
    TabellenZeiger : POINTER TO CARDINAL;
(*$S- $F- $E-*)
PROCEDURE DaysInYear;
BEGIN
   INLINE(  0,  31,  59,  90, 120, 151, 181,
          212, 243, 273, 304, 334);
END DaysInYear;
   HilfsZeiger^    := DaysInYear;
   TabellenZeiger := CAST(ADDRESS, HilfsZeiger^);
```

Listing 1. Maschinennahes Anlegen einer Tabelle ist bereits während des Compilierens möglich

dicht, sehr schnell und ohne überflüssiges »overhead«,
— der Linker wird entlastet,
— der Objektcode wird kürzer
— und unnötige Diskettenzugriffe unterbleiben.

Wir stellen ein vergleichendes Beispiel in Listing 3 vor das den Vorteil der neuen Technik des »m2c« erkennen läßt.

Dem Beispiel ist zu entnehmen, wie unterschiedlich die Compiler »Aztek-C« und »m2c« den Quelltext der ersten Zeile verarbeiten. Für den Aufruf der Systemroutine »Draw« benötigt der von Aztek-C erzeugte Code 138 Zyklen und der von »m2c« erstellte 90.

Weitere Analysen bestätigen, daß der von »m2c« erzeugte Code Systemroutinen stets schneller aufruft. Das System kann aber noch weiter verbessert werden. Die Untersuchung des Codes (siehe Listing 3) ergibt, daß überflüssige Stackoperationen (zum Beispiel move.w #$000a,-(A7) und move.w (A7)+,D1 anstelle von move.w #$000a,D1) im angeführten Objekt-Code wertvolle 24 Taktzyklen verschwenden.

Das ist Kunst

Der Compiler kann über zehn Optionen gesteuert werden. Alle sind dokumentiert. Es wird jedoch nicht erwähnt, wie mehrere Optionen zusammengefaßt werden, und unter welchen Bedingungen Optionen mit Kommentaren zu mischen sind. Die Dokumentation würde an Wert gewinnen, wenn sie Aussagen über den Zeitbedarf für Kontrollen während des Programmablaufs enthielte und präziser ausführte, in welchen Fällen Code zur Durchführung von Kontrollen generiert wird. Durch Zufall entdeckten wir zum Beispiel, daß vor Aufruf einer Systemroutine der noch vorhandene Platz auf dem Stack geprüft wird. Ein Zeichen, daß die Compilerbauer ein Werkzeug für möglichst sicher arbeitende Software entwickeln wollten.

Der Ablauf der mit »m2c« erstellten Programme wird durch das »Amiga Runtime System« (Arts) vorzüglich unterstützt. Es umfaßt entgegen der Dokumentation 4,5 und nicht 2,5 KByte. Die Abkürzung »Arts« hat nichts mit Künsten zu tun, sondern eher mit einer Abneigung gegen Gurus. Und damit sind wir auch schon bei dem ersten wesentlichen Vorteil von »Arts« gegenüber anderen Laufzeitsystemen: Es fängt Programmabstürze ab. Das System hat mehrfach bewiesen, daß es Laufzeitfehler (zum Beispiel Division durch 0) sicher erkennt: Sofort schließt es offene Dateien und gibt allozierte Speicherbereiche frei. Darüber hinaus erlaubt »Arts«:
— die Ausführung von Programmen mit <Control C> abzubrechen,
— »BreakPoints« zu setzen,
— den Programmablauf von bestimmten Bedingungen abhängig zu machen und den Grund für den Abbruch anzuzeigen, und
— den Debugger »Assert« einzubinden sowie Abschlußprozeduren zu installieren.

3.40	3.00	Aztek-C 3.1 d	TDI-Modula	M2Amiga
Compilieren	NumbTest	18	20	02
(sec)	Primes	14	17	02
Linken	NumbTest	46	25/19	03
(sec)	Primes	26	07/07	03
Länge des	NumbTest	6352	3048/8380	7884
Compilats	Primes	1928	1280/4036	4668
Laufzeit	NumbTest	13	23/15	13
(sec)	Primes	13	14	13

Tabelle. »M2Amiga« schneidet im direkten Vergleich mit »Aztek-C« und »TDI-Modula« ganz hervorragend ab

```
"Aztek-C"               "TDI-Modula"              M2Amiga-Modula"
clr.w  -2(a5)           CLR.W   0000FA02           clr.w  -4(A4)
loop:
                                                   cmpi.w  #$2710,4(A4)
                                                   bhi.l   loopend
move.w -2(a5),d3        MOVEQ   #0003,D5           move.w -4(A4),D6
mulu   %3,d3            MULU    0000FA02,D5        mulu    #$3,D6
move.w d3,-4(a5)        MOVE.W D5,0000FA04         move.w D6,-6(A4)
add.w  %1,-2(a5)        ADDQ.W  #1,0000FA02        addq.w #1,-4(A4)
cmp.w  %10000,-2(a5)    CMPI.W  #2711,000FA02
blt    loop             BNE     loop               bcc.s loop
loopend:
```

Listing 2. Die Codegenerierung für eine FOR-Schleife

```
            Aztek-C                           M2Amiga-Modula

Quelltext: Draw(rp, 10, 20);            Draw(P, 10, 20);

Assembler:                              move.l 6(A4),A1
           move.w   #20,-(a7)           move.w  #$000a,-(A7)
           move.w   #10,-(a7)           move.w  #$0014,-(A7)
                                        move.w (A7)+,D1
                                        move.w (A7)+,D0
           move.l   _rp,-(a7)           move.l -70(PC),A6
                                        jsr    -246(A6)
           jsr      _Draw
           add.w    #08,a7
_Draw:     move.l    4(sp),a1
           movem.l  8(sp),d0/d1
           move.l    _GfxBase,a6
           jmp       -246(a6)
```

Listing 3. Auszugsweiser, disassemblierter Code für den Aufruf der Systemprozedur »DRAW«, erzeugt von »Aztek-C« und »M2Amiga«. M2Amiga ist Sieger nach Punkten.

```
(* $M- *) (* ohne Implementation*)
DEFINITION MODULE CodeTest;

FROM SYSTEM IMPORT ADDRESS;
PROCEDURE P1 (a6[14] : ADDRESS):LONGINT;
   CODE -40;
END CodeTest.
```

Listing 4. Der Befehl »CODE« erzeugt einen Sprung mit indirekter Adressierung (mit Distanz): JSR -40,(A6)

Das Konzept der Abschlußprozeduren erlaubt es, parameterlose Prozeduren bei »Arts« anzumelden, die im Falle eines Abbruchs oder bei normaler Beendigung eines Programms abgearbeitet werden. Selbst bei Programmabbrüchen erreichen Sie einen definierten Zustand. Auch dies dürfte bei Programmierern nicht unbeachtet bleiben. Das Laufzeitsystem übernimmt es sogar »Amiga-Libraries« zu öffnen und zu schließen.

Über den Compiler ließe sich sicher noch viel mehr schreiben — vermutlich ein ganzes Buch. Wer mehr darüber wissen möchte, der sollte sich die Test- beziehungsweise Demoversion auf der Fish-Disk 113 anschauen. Nähere Informationen hierzu finden Sie auf Seite 103. Doch verlassen wir die Besprechung des Compilers, wenden wir uns dem Linker zu.

Sekundenkleber

Der Linker besticht wie der Compiler durch sein hohes Arbeitstempo. Dies liegt auch daran, daß er die importierten Module vollständig bindet und nicht nur die tatsächlich benutzten Objekte. Fertige Programme sind daher meistens länger, als die mit Aztek-C oder TDI-Modula erzeugten. Der Nachteil wirkt sich jedoch nur gravierend aus, wenn der Code in EPROMS Platz finden soll. Die Entwickler des »m2l« werden den Wunsch verstehen, den Linker mit einer zweiten Betriebsart auszustatten, die den Code optimiert.

»M2Amiga« enthält eine umfassende und zuverlässig arbeitende Standardbibliothek. Sie folgt mit wenigen Ausnahmen den Empfehlungen von N. Wirth [1]. Einige Bezeichner wurden geändert und mehrere Prozeduren anders verteilt. Ein Modul namens »STREAMS« fehlt. Die Dokumentation ist allerdings für Einsteiger nicht ausreichend. Weitere Beispiele wären von Vorteil.

Da die Module teilweise recht lang sind, sollten Sie vor jedem Import prüfen, ob es günstigere Möglichkeiten gibt. So werden einfache Ein- und Ausgaben von Text durch das Modul »Terminal« mit wesentlich weniger unbenötigtem Code belastet, als durch das Modul »InOut«. Hervorzuheben ist, daß »Terminal« selbständig ein Fenster für die Ein- und Ausgabe öffnet, sofern das importierende Programm von der Workbench gestartet wird.

Sehr praktisch ist auch das Modul »Windows«, das eine einfache aber leistungsfähige Schnittstelle bietet, um mit Fenstern zu arbeiten. Als unpraktisch dagegen erweist sich, daß die »Read-Prozeduren« der Module »InOut« und »Terminal« nicht auf Tastendruck ein Zeichen einlesen, sondern zusätzlich <RETURN> bedürfen. Der Grund besteht darin, daß beide Module in Fenstern des Typs »CON:« arbeiten und nicht »RAW:« verwenden. Ansonsten wird der Amiga gut unterstützt.

Die Standardbibliothek wird durch 32 Module ergänzt, die als Schnittstellen zum Betriebssystem des Amiga realisiert sind. Die mitgelieferte Dokumentation enthält im wesentlichen nur Definitionen beziehungsweise Deklarationen zu den verwendeten Objekten. Ihre Wirkungsweise ist der Dokumentation zum Amiga [3] zu entnehmen. Einsteiger werden es nicht leicht haben, mit der derzeitigen Dokumentation ihren Amiga in den Griff zu bekommen. Erschwerend wirkt, daß die ROM-Kernel-Manuals ausschließlich auf C abgestimmt sind und noch nicht einmal der aktuellen Version des Betriebssystems entsprechen. Erfreulicherweise weichen die vom M2Amiga verwendeten Bezeichner nur selten von der offiziellen Dokumentation ab, so daß von dieser Seite keine Orientierungsschwierigkeiten eintreten. Im knappen Handbuch liegt eigentlich der einzige bedeutsame Schwachpunkt des Programmiersystems M2Amiga.

In [2] haben wir bereits Compiler für den Amiga verglichen. Den dort dargestellten Testbedingungen haben wir auch M2Amiga unterzogen: Die Ergebnisse sind in der Tabelle dargestellt.

Bei der Interpretation der Ergebnisse fallen insbesondere die sehr kurzen Zeiten für das Compilieren und Linken auf. Die Ausführungszeit der Programme steht der der mit »Aztek-C« erzeugten um nichts nach. Beide Tests sind allerdings nicht geeignet, die Stärken von »M2Amiga« beim Aufruf von Systemroutinen praktisch zu belegen.

»M2Amiga« ist ein attraktives Werkzeug zur Erstellung anspruchsvoller Software. Das Entwicklungssystem arbeitet zuverlässig und effizient. Die einzelnen Komponenten sind auf der Ebene der »Workbench« beispielhaft integriert. Die Dokumentation ist in Teilen stark ergänzungsbedürftig. Bei der Arbeit mit »M2Amiga« stellt sich schnell ein Gefühl der Sicherheit ein, das vor allem vom Compiler und dem exzellenten Laufzeitsystem ausgeht. Der Linker sollte aber auf jeden Fall nur tatsächlich benötigte Objekte binden. Mit »M2Amiga« erwerben Sie eine saubere Implementation von Modula-2 auf dem Amiga. Es macht Spaß, damit zu arbeiten.

(Edgar Meyzis/ub)

Quellen:

[1] N. Wirth, Programmieren in Modula-2, 3. überarbeitete Auflage, Berlin, 1985

[2] R. Beaupoil, Schnelle Compiler, AMIGA, Ausg. 6/7-87, S.90 ff.

[3] Amiga Intuition Reference Manual, 1986
Amiga ROM Kernel Reference Manual: Libraries and Devices,1986
Amiga ROM Kernel Reference Manual: Exec, 1986

AMIGA-WERTUNG

Software: M2Amiga

8,9 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■			
Bedienung	■	■	■	■	■	
Leistung	■	■	■	■	■	

Fazit: M2Amiga ist ein schneller Modula-2-Compiler für den Amiga. Die Laufzeiten der fertigen Programme können sich mit C-Programmen messen. Es wird interessant sein zu beobachten, ob Modula-2 sich mit diesem Computer auf dem Amiga etablieren wird. Die Voraussetzungen dafür sind günstig.

Positiv: kein Kopierschutz; Editor erlaubt das schnelle Auffinden von Fehlern im Quell-Code; hervorragendes Laufzeitsystem »Arts«; schneller Linker; mathematisch saubere Behandlung von DIV und MOD; Sprachumfang berücksichtigt bereits sich abzeichnende Entwicklungen.

Negativ: Dokumentation noch zu dürftig; Read-Prozeduren benötigen die zusätzliche Eingabe von <Return>; CAST-Anweisung noch eingeschränkt.

DATEN

Produkt: M2Amiga

Preis: 342 Mark

Hersteller: A.+L.Meier-Vogt, Im Späten 23, CH-8906 Bonstetten/Zh, Tel. 01/7003037, Intern. 0041-1-7003037

Diskettenstation im Eigenbau

Zweitlaufwerke sind immer noch relativ teuer. Sparen Sie Geld und bauen sich selbst mit wenig Aufwand ein (nötiges) Zusatzlaufwerk für Ihren Amiga.

Der Amiga — ein Traum? Ja natürlich, werden jetzt viele sagen. Er besitzt überragende Grafik-, Sound- und Programmiermöglichkeiten. Und er besitzt, da fängt das Erwachen schon an, in der Grundausstattung nur ein Diskettenlaufwerk, was zum effektiven Arbeiten zu wenig ist. Wer kennt nicht das Problem, bei dem Versuch, beispielsweise das Directory einer anderen Diskette zu lesen, die Antwort »Please insert Workbench...« zu erhalten. Oder die Probleme bei Programmen, die nur mit zwei Laufwerken arbeiten (etwa einige Kopierprogramme). So wird bald der Wunsch nach einem Zweitlaufwerk wach. Nur fehlt oft das Geld, das Laufwerk auch zu bezahlen. Hier hilft der folgende Artikel weiter, um mit etwas Lötarbeit, ein wenig basteln und zu einem akzeptablen Preis (meist unter 200 Mark) doch noch zu einem Zusatzlaufwerk zu kommen.

Die Elektronik

Die Schaltung selektiert das angeschlossene Laufwerk, bearbeitet den Diskettenwechsel, ist abschaltbar und kann auf die Laufwerksnummer DF1, DF2 oder DF3 eingestellt werden. Darüber hinaus ist sie doch so klein gehalten, daß die Platine in jedes normale Floppygehäuse noch zusätzlich eingebaut werden kann. In Bild 1 sehen Sie den Schaltplan der Controller-Schaltung.

Wen die folgende Erklärung der Elektronik nicht interessiert, kann diesen Teil überspringen. Vorab sei gesagt, daß sowohl der Bus auf der Amiga-Seite als auch der Shugart-Bus des Laufwerks (verwendet wird ein NEC 1036 A-Laufwerk) LOW-aktiv sind. Das bedeutet: nur wenn an einem Pin ein Signal anliegt, dann geht der Pegel an diesem Pin auf 0 Volt. Ansonsten liegen alle Pegel auf HIGH.

Nach einem Systemreset wird »DRes« (Drive Reset) vom Amiga auf Null-Pegel gezogen. Dadurch werden die Motoren aller angeschlossenen Laufwerke abgeschaltet und die Laufwerke in einen definierten Grundzustand gebracht. Anschließend schaltet das System die Select-Leitungen (Sel) und »Mtrx« (Motor on) auf LOW. Normalerweise bewirkt »Mtrx«, daß alle Laufwerke ihre Motoren anschalten. Bei mehreren angeschlossenen Laufwerken (maximal vier) ist das etwas ungeschickt. Ein Flipflop übernimmt deshalb die Selektierung. Es ist in der Lage, ein Bit zu speichern. Hier wird ein positiv-flankengetriggertes D-Flipflop verwendet.

Das Signal der voreingestellten Sel-Leitung (mit den kleinen Schaltern kann man die Laufwerknummer einstellen — Pin 21 = DF1, Pin 9 = DF2, Pin 20 = Df3) wird invertiert und das Mtrx-Signal ins Flipflop übernommen. Über ein NAND-Gatter wird der Ausgang an den Ready-Pin gemeldet. Da der Amiga die Selx-Leitungen nacheinander abfragt, erkennt er nun aufgrund der jeweiligen Ready-Rückmeldung, welches Laufwerk angeschlossen ist und richtet den entsprechenden Speicherplatz im Speicher ein. Diese Prozedur läuft bei jedem Schreib-/Lesezugriff ab. Da auch die Mto- (Motor on) und die Inu-(In use)Leitungen über ein NAND-Gatter an den Ausgang des D-Flipflops angeschlossen sind, können die Motoren der Laufwerke unabhängig voneinander ausgeschaltet werden.

Die nächste Schwierigkeit liegt im Chng-Signal (Disk-Change). Dieses Signal stellt das NEC-1036A-Laufwerk nicht am Shugart-Bus zur Verfügung. Es liegt im Laufwerk an einem Lötstützpunkt an. Von dort wird es über einen Draht an P1 auf die Zusatzplatine eingespeist. Auch hier muß man dem Amiga verständlich machen, bei welchem Laufwerk ein Diskwechsel vorliegt. Dies geschieht durch die Verschaltung mit dem jeweiligen Select-Signal. Der Computer erkennt nun einwandfrei, wo eine Diskette gewechselt wurde und initialisiert das entsprechende Laufwerk neu.

Als weiteres Bonbon bietet die Schaltung die Möglichkeit, das Laufwerk für den Amiga »wegzuschalten«. Wenn in einem angeschlossenen Laufwerk keine Diskette eingelegt ist, dann meldet es dies dem Amiga durch ein LOW-Signal auf der Chng-Leitung. Die Leitung bleibt auf LOW, bis der Computer einen Step-Impuls auslöst (auf Pin 18). Ist zu diesem Zeitpunkt schon wieder eine Diskette im Laufwerk, springt Chng auf HIGH-Pegel zurück und der Amiga initialisiert. Ansonsten bleibt die Leitung auf LOW und der Computer muß weiterhin in regelmäßigen Abständen einen Step-Impuls auslösen, um zu erkennen, ob sich wieder eine Diskette im Laufwerk befindet. Mit dem Schalter S1 wird die Select-Leitung unterbrochen, worauf der Computer das Laufwerk nicht mehr erkennt, die Knackgeräusche unterbleiben. Man beachte aber, daß der für das Laufwerk reservierte Speicherplatz dem System erst nach einem Reset wieder zur Verfügung steht.

Zusammenbau

Nach dem Herstellen der Interface-Platine (das Layout sehen Sie in Bild 2) und dem Besorgen der Bauteile (Tabelle 1) kann mit dem Aufbau begonnen werden. Sehen Sie sich dazu den Bestückungsplan in Bild 3 an. Als erstes lötet man mit Hilfe des blanken Silberdrahtes die eingezeichneten Lötbrücken ein. Anschließend werden die ICs, dann die Widerstände, der Kondensator und die DIP-Schalter sowie die Flachstecker eingelötet. Bei den Flachsteckern sollte darauf geachtet werden, daß sie wirklich gerade auf der Platine sitzen, da sonst Kontaktprobleme entstehen können.

Isolieren Sie danach das Rundkabel am einen Ende etwa fünf Zentimeter und am anderen Ende etwa acht Zentimeter ab. Man entfernt die Isolationslitze bis auf einen kleinen Rest, verdrillt diesen und verzinnt ihn. Er dient als Masseleitung. Am D-Sub-Stecker werden die Pins 3 bis 7 mit einem blanken Draht verbunden und an die Masseleitung des Rundkabels angeschlossen. Die restlichen Leitungen verbinden Sie wie aus Bild 1 ersichtlich. Die jeweiligen Pinnummern in der Zeichnung stimmen mit denen des Steckers überein. Bohren Sie in der Rückblende des Gehäuses ein entsprechendes Loch für den Schalter. Stecken Sie nun die Zugentlastung und die Gehäuserückwand lose auf das Kabel. Dies ist notwendig, da das Laufwerk nur von einer Seite in das Gehäuse eingebaut werden kann. Aber Vorsicht: Zuerst planen, dann bauen. Man spart sich dadurch viel Arbeit. Im nächsten Schritt wird das Kabel an der Platine festgelötet. Beachten Sie hierbei unbedingt die Farben der einzelnen Leitungen, damit gewährleistet ist, daß wirklich die entsprechenden Pins des Steckers mit den zugehörigen Lötaugen auf der Platine verbunden werden!

Zur Erzeugung des Chng-Signals wird im Laufwerk am Pin OUT der isolierte Draht angelötet. Dieser Pin befindet sich neben der Laufwerks-LED, oberhalb der großen Motorschwungscheibe. Den Draht verlegt man sorgfältig so durchs Laufwerk bis zum Stecker, daß der Motor nicht behindert wird. Zum Schluß verlöten Sie noch die zwei isolierten Drähte für den einpoligen Ausschalter auf der Platine.

An den DIP-Schaltern stellt man die gewünschte Laufwerksnummer ein und steckt die Platine samt Kabel von der einen Seite durch das Gehäuse auf das Laufwerk. Verlöten Sie noch das isolierte Kabel des Chng-Signals mit der Platine und schieben das Laufwerk in das Gehäuse, welches anschließend verschraubt wird.

Ein Hinweis noch: Die Schaltung dieser Platine wurde nach bestem Wissen entwickelt und ausführlich getestet. Der Autor sieht sich aber gezwungen, darauf hinzuweisen, daß keine Garantie übernommen wird und die Schaltung ohne Rücksicht auf die Patentlage weitergegeben wird.

Sollten Schwierigkeiten beim Besorgen der Bauteile auftreten, bietet der Autor seine Hilfe an. Über ihn können Sie vor allem die begehrten Flachstecker anfordern, da sie leider nicht so leicht im Fachhandel zu bekommen sind. Den Autor können Sie unter folgender Anschrift erreichen:

Gerhard Stock
Passauerstr. 35b
8500 Nürnberg 30
Tel. 09 11/40 36 81

(Gerhard Stock/dm)

One step ahead !

AMIGA·SHOP·DORTMUND

IS JA IRRE !

Anwendungs-Software	Preis DM
Aegis Animator & Images	249,--
Aegis Draw Plus	489,--
Aegis Impact	179,--
Aegis Sonix	198,--
Butcher V2.0	67,--
CLI-Mate V1.2	67,--
Deluxe Music (deutsch)	197,--
Deluxe Paint II Art Parts2	29,--
Deluxe Paint II Seasons +	29,--
Deluxe Paint II (deutsch)	245,--
Deluxe Print II (deutsch)	197,--
Deluxe Video 1.2 (deutsch)	249,--
Devpac Assembler	148,--
Go Amiga Datei	179,--
Prism	159,--
Sculpt 3D	179,--
Superbase Amiga	249,--
VideoScape 3D (deutsch)	385,--
Vizawrite	198,--
Word Perfect V4.1	698,--
Zing	147,--
Zing Keys	87,--

Bücher	Preis DM
Amiga 2000 Buch	59,--
Amiga 500 Buch	49,--
Amiga Assembler Buch	59,--
Amiga C in Beispielen	69,--
Amiga DOS Handbuch	59,--
Amiga Programmierhandbuch	69,--
Deluxe Grafik auf d. Amiga	49,--
Programmieren mit A.-Basic	59,--

Hardware	Preis DM
Amiga 2000	2295,--
Amiga 2000 & Monitor 1084	2895,--
Amiga 500	1048,--
Druckerkabel A 500/ A 2000	34,95
Echtzeit-Digitizer	1749,--
Fuji-Disks 3,5" 1DD 10Stk.	25,95
Fuji-Disks 3,5" 2DD 10Stk.	29,95
Genlock-Adapter	1799,--
Monitor 1081	679,--
Monitorkabel	24,95
Mouse-Pad	17,90
NEC 1036A LW extern kompl.	349,--
NEC 1036A LW intern kompl.	289,--
NEC Multisync Monitor	1399,--
NEC-Drucker P6	1098,--
NEC-Drucker P7	1498,--
Speichererweiterung A 501	279,--
Speichererw. 2MB ext. A500	899,--
Speichererw. 512k ohne Uhr	199,--
Staubschutzhaube für A 500	24,95

Spiele	Preis DM
Amegas	59,--
Bad Cat	57,--
Barbarian (Psygnosis)	67,--
Bards Tale I	99,--
Black Cauldron	79,--
California Games	79,--
Chessmaster 2000	77,--
Emerald Mine	27,--
Feary Tale	99,--
Feud	27,--

Fire Power	79,--
Flightsimulator II	109,--
Garrison	69,--
Garrison II	69,--
Goldrunner	77,--
Great Giana Sisters	59,--
Grid Start	27,--
Gunship	75,--
Jagd auf Roter Oktober	69,--
Jinks	59,--
Jinxter	87,--
Karate Kid II	67,--
King of Chicago	87,--
Knight Orc	67,--
Land of Legends	89,--
Las Vegas	27,--
Leviathan	67,--
Marble Madness	67,--
Mouse Trap	59,--
Phalanx	27,--
Phalanx II	27,--
Pinball Wizard	47,--
Q-Ball	69,--
Rallye Master	29,--
Rasterbike	29,--
Soccer King	27,--
Space Ranger	27,--
Street Gang	59,--
Terrorpods	67,--
Test Drive	79,--
Tetris	67,--
Willy the kid	27,--
Wizball	69,--

Lieber Sekt saufen und sofort bestellen, als abwarten und Tee trinken.

©'88

Rezeptfrei im Direktversand per Nachnahme oder Vorkasse (+ Porto/Verp.)

A·S·D
Knack Computertechnik

Leistungsstark!

Wilmsmannstraße 14
4600 Dortmund 30
☎ 0231/485567

Fordern Sie unsere kostenlose, aktuelle Liste an !

Präsident Printer 6320

compatible zu fast allen Computern

Zeichensätze:
Epson*-Commodore*
Epson*-Centronics*
Epson*-V 24/RS 232 C
100 % Commodore compatible, Commodore* Befehlssatz, 64er + 128er Zeichensatz
Amiga*-Zeichensatz
IBM* Befehls- und Zeichensatz 1 + 2
Schneider* Befehls- und Zeichensatz
Atari* ST-Zeichensatz
Thomson* Befehlssatz
TA*-Zeichensatz
9 internationale Zeichensätze

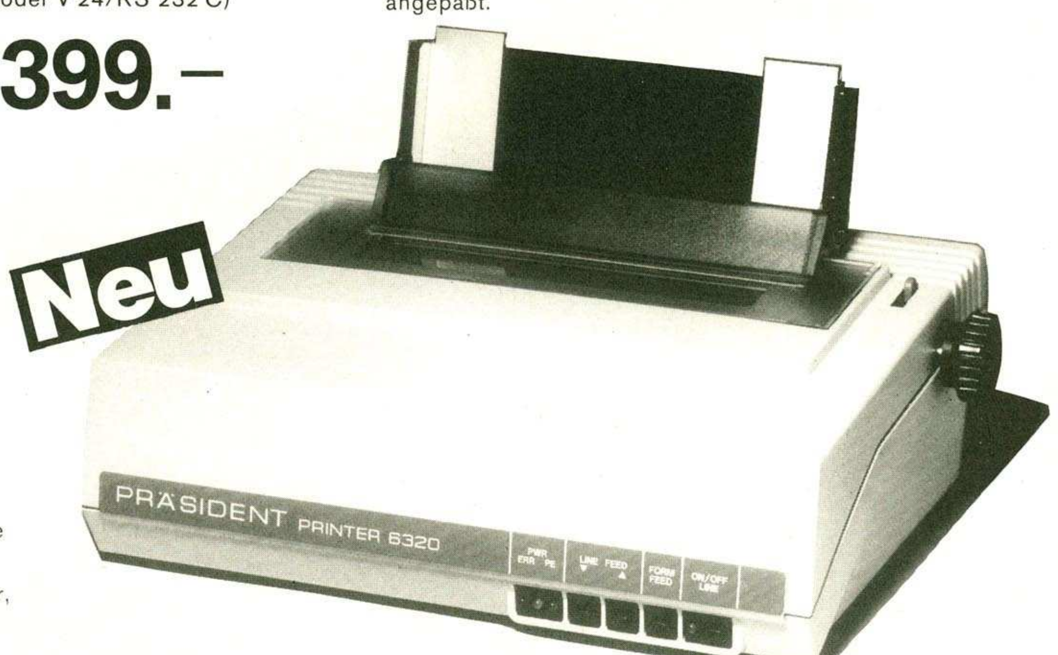
Technische Daten:
- 100 Zeichen pro Sekunde
- Druckmatrix 9 x 9
- Hochauflösende Grafik mit 480, 576, 640, 720, 960 und 1920 Punktdichte pro Zeile
- Logisch druckwegoptimiert, bidirektional
- Schriftarten: Pica, Elite
- Druckarten: Normal, doppelt, breit, komprimiert, Sperrschrift, Exponenten/Indices, automatisches Unterstreichen.
- NLQ (schreibmaschinenähnliche Druckqualität)
- Verstellbare Stachelradwalze für randgelochtes Endlospapier, Einzugsschacht für Einzelblatt sowie Staubschutzhaube im Preis inbegriffen.

unverbindliche Preisempfehlung incl. Interface (wahlweise Centronics*, Commodore* oder V 24/RS 232 C)

399.–

Computerwechsel:
Interface tauschen, schon ist der Präsident Printer 6320 angepaßt.

Horst Grubert,
Import · Agentur
8110 Murnau



*Atari, Centronics, Commodore, Amiga, Epson, IBM, Schneider, Thomson sind eingetragene Warenzeichen der Atari Corp., Centronics Data Computer Corp., Commodore Corp., IBM Corp., Schneider GmbH, Thomson Corp.

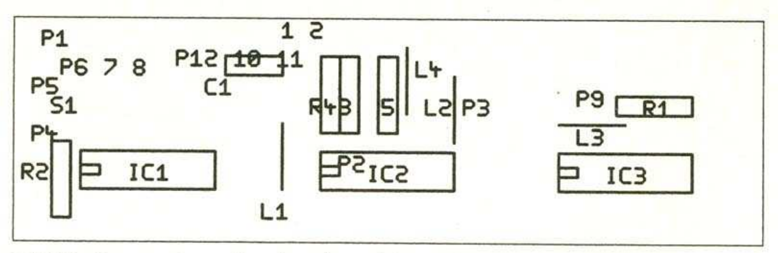


Bild 3. So werden die einzelnen Bauteile auf die Platine gelötet

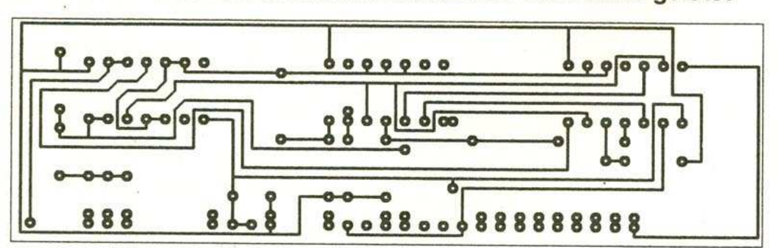


Bild 2. Das Platinenlayout der Interfaceplatine im Maßstab 1:1 (spiegelverkehrt)

IC1 — 74LS00
IC2 — 74LS74
IC3 — 74LS38
R1 bis R5 — 5 Widerstände 1 kΩ, Metallfilm, 0,25 Watt
C1 — Kondensator, 100 nF, 50 V, Keramik
S1 — einpoliger Ein-/Ausschalter
1 23poliger D-Sub-Stecker mit Plastikgehäuse
1 einreihiger Flachstecker, 17polig, zirka 10-14 mm hoch
1 einreihiger Flachstecker, 4polig, zirka 10-14 mm hoch
1 dreifacher DIP-Schalter
etwa 70 cm Rundkabel, 20adrig, abgeschirmt
ein zum Laufwerk passendes Metallgehäuse nach Wahl
etwa 30 cm isolierte Kupferlitze
etwa 20 cm blanken Silberdraht
1 Platine wie aus Bild 2 ersichtlich

Tabelle. Diese Bauteile werden zum Bau des zusätzlichen Amiga-Diskettenlaufwerks benötigt

Amiga 23-D-Sub-Stecker
NEC 1036a-34-Pin-Connector

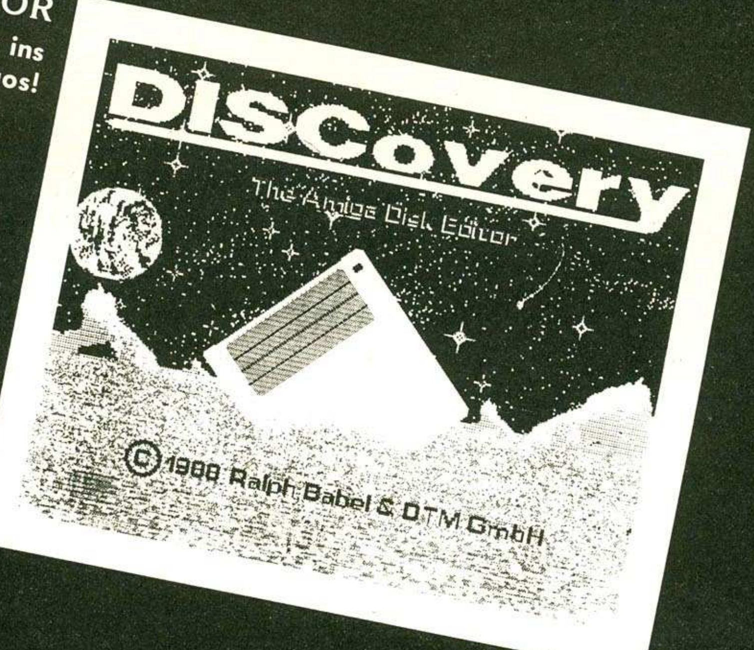
1 — Ready — 34
13 — Side-Select — 32
2 — Disk Read Data — 30
14 — Write Protect — 28
15 — Track Null — 26
16 — Disk Write Enable — 24
17 — Disk Write Data — 22
18 — Step — 20
19 — Direction — 18
22 — Index — 8
10 Drive Reset — P3
8 Motor X — P2
20 Select 3 = DF3
9 Select 2 = DF2
21 Select 1 = DF1
11 Disk Change — P9
3 GND — P10
4
5
6
7
P6 P7 P8
Federschalter P5
S1
Masse
P4
+5V R2
R3 +5V R5 +5V
16
14 NC
12 NC
10
6 NC
4
2 NC
D S Q
T
R Q̄
Flip-Flop
N1 N2 N3 N4 N5
+5V
12 — P11
23 — P12
+12V
R4
+5V
an sämtliche unbenützte Eingänge aller Ic's, um Schwingungen zu vermeiden
Anschluß für interne Floppyleitung — P1
P9
+5V R1
+12V
P12 — 4
P10 — 3
2
P11 — 1
Masse +5V

Bild 1. Der vollständige Schaltplan für die zugehörige Interfaceplatine

DISCOVERY
THE AMIGA DISK EDITOR

.... **Ordnung ins**
.... **Diskettenchaos!**

- Totale Diskettenkontrolle
- Automatische Fehlerdiagnose
- Viruserkennung
- Menüführung & Maussteuerung
- Repariert Lesefehler
- Wiederherstellung gelöschter Daten
- Änderung fremder Programme
- Flexible Daten-Ein- und Ausgebe
- Komplexe Suchfunktionen
- Volle Drucker-Unterstützung
- Kein Kopierschutz



Ausführliches Deutsches Handbuch (200 Seiten) mit Einführung in die Disketten-organisation und Referenzwerk von Ralph Babel
Unverbindl. Preisempfehlung... **198.00**
(Kostenloses Demo gegen Einsendung einer Leerdiskette mit frankiertem Rück-umschlag erhältlich.)

FUNKTION

.... **Perfekte Funktionsanalyse**

- 50 Funktionen gleichzeitig darstellbar
- Ausblenden einzelner Funktionen
- Flexible Parametereingabe
- Koordinatensystem automatisch oder manuell
- Berechnung von Ableitungen
- Alle Auflösungen bis 32 Farben wählbar
- Speichert und druckt Funktionen
- Sehr schnelle Zeichenroutine
- Menüführung in deutsch & Maus-gesteuert
- Kein Kopierschutz

Funktion für alle Schüler, Studenten, Wissenschaftler und Entwickler. Aus-führliches deutsches Handbuch von Mi-chael Schreiner.
Unverbindl. Preisempfehlung.... **98.00**
(Kostenloses Demo gegen Einsendung einer Leerdiskette mit frankiertem Rück-umschlag erhältlich.)

AEGIS-Produkte und andere DTM AMIGA-Software erhalten Sie u.a. bei: WAW Eletronic, Tegeler Str. 2, 1000 Berlin; Boysen & Maasch, Hermannstr. 31, 2000 Hamburg; Hamburger Softwareladen, Gärtnerstr. 5, 2000 Hamburg; System Shop, Lilienstr. 23, 2000 Hamburg; Reese GmbH, Rendsburger Landstr. 196, 2300 Kiel; GOLD Computerhaus, Donnerschweerstr. 127, 2900 Oldenburg; CSJ Computersoft Jonigk, An der Tiefenriede 27, 3000 Hannover; Diamond Soft, Regentstr. 178, 4050 Mönchengladbach, Softshop, Müllergasse 6-8, 4100 Duisburg; International Software Heidenrichstr. 10, 5000 Köln 80; MRZ, Kölner Landstr. 240, 5160 Düren; Hansen & Gieraths EDV, Münsterstr. 1, 5300 Bonn; Kaurisch GmbH, Zuckerbergstr., 5500 Trier; Mediencenter, Wermingser Str. 45, 5860 Iserlohn; Intelligent Memory, Basaltstr. 58, 6000 Frankfurt; Computer Landolt, Wingertstr. 114, 6457 Maintal 1; Kaurisch GmbH, Trierer Str. 16-22, 6600 Saarbrücken; SHOP 64, Großherzog Friedrich-Str. 40, 6600 Saarbrücken; SHOP 64, Titzstr. 24, 6630 Saarlouis; SHOP 64, Talstr. 44, 6650 Homburg; SHOP 64, Ludwigstr. 46, 6670 St. Ingbert; Proficenter, Lutherstr. 7, 6680 Neunkirchen; SHOP 64, Bellesweiler Str. 13, 6680 Neunkirchen; Profisound, Schillerstr. 36, 6700 Ludwigshafen; Kaurisch GmbH, Richard Wagner-Str. 20-22, 6750 Kaiserslautern; Gausch & Sturm, Casterfeldstr. 74, 6800 Mannheim; Computerstudio, Hasenbad 18/1, 7900 Ulm; Musik & Grafiksoftware, Wasserburger Landstr. 244, 8000 München; Technoland, Werftstr. 15, 8360 Deggendorf; Technoland, Neuburger Str. 104, 8390 Passau; Technoland, Lilienthalstr. 3, 8400 Regensburg; Technoland, Rothenburger Str. 451, 8500 Nürnberg; Schöll Büroteam, Dominikanerplatz 5, 8700 Würzburg; Bissinger Bürotechnik, Industriestr. 18, 8883 Gundelfingen; Adolf Schmoll Computerstudio, Schwalbenstr. 1, 8900 Augsburg sowie bei **HAKO AG,** Photo, Video, Elektronik, **SATURN HANSA** und anderen guten Fachhandelsgesellschaften. Vertriebsrechte für die BRD: **DTM Werbung und EDV GmbH,** Poststr. 25, 6200 Wiesbaden-Bierstadt, (0 61 21) 56 00 84, fax (0 61 21) 56 36 43.

MICROTRON
COMPUTERPRODUKTE
Postfach 69 Bahnhofstr. 2
Tel. 032 872429 Fax 032 872482
CH-2542 PIETERLEN



E&S–WI

Der Weg in

Mit einer Ein-/Ausgabekarte steht dem Amiga der Weg nach außen offen. Mit unserer Bauanleitung eröffnen Sie auch Ihrem Amiga neue Möglichkeiten der Entfaltung durch Schaffung neuer Ports.

Alle Anwender, die vom Commodore 64 auf den Amiga umgestiegen sind, und alle, die Peripheriegeräte an den Amiga anschließen wollen, stehen einem kleinen Problem gegenüber: Der sonst sehr leistungsfähige Computer stellt keine gleichwertige Schnittstelle zur Außenwelt zur Verfügung wie beispielsweise der C 64 mit seinem User-Port. Dies läßt sich jedoch mit der im Anschluß vorgestellten VIA-Karte beheben, die folgende Eigenschaften besitzt:

— 36 frei programmierbare I/O-Ports
— 4 universelle Timer
— 2 Schieberegister
— 4 Handshakeleitungen zum Datenverkehr
— Interruptfähig
— einfach in den Expansion-Port des A 500 einsteckbar
— keine Belegung von Speicherbereichen
— voll kompatibel zur übrigen Hardware
— geringe Kosten durch Selbstbau (unter 100 Mark)

Dies alles findet auf einer Platine mit den Abmessungen 100 x 120 mm Platz, wobei der Nachbau durch das abgebildete Layout vereinfacht wird.

Die Platine (durchkontaktiert mit Lötstopmaske) kann bei der am Schluß stehenden Adresse zum Preis von 30 Mark bezogen werden. Ebenfalls ist dort der komplette Bausatz (90 Mark) sowie das Fertiggerät (115 Mark) zu beziehen.

Die Platine kann Verwendung in vielen Anwendungen finden. Hier nur eine kleine Auflistung der Möglichkeiten: Alarmanlage, Logikanalysator, Schrittmotorsteuerung oder Lichtorgel.

Es folgt eine Beschreibung der Hardware anhand des aus Bild 1 ersichtlichen kompletten Schaltplans:

Die internen CIAs des Amiga, welche die Steuerung des Diskettenlaufwerks, der Tastatur und der weiteren Schnittstellen erledigen, belegen im Speicher den Adreßbereich von $BFD000 bis $BFE000. Da diese aber nur mit den Adreßleitungen »A12« und »A13« selektiert werden, erscheinen sie im Speicher im ganzen Bereich von $A00000 bis $BFFFFF jeweils auch gespiegelt, da GARY (der Decoderbaustein im A500) dem Prozessor in diesem Bereich mitteilt, daß dieser im synchronen Modus arbeiten soll. Da der Prozessor als Folge davon seine Leitung »-VMA« auf Low-Pegel legt, welche ebenfalls zur Decodierung verwendet wird, entsteht diese Spiegelung der CIAs im Speicher.

Der Grundstock: die Hardware

Hier wurde nun eine Möglichkeit gesucht, zusätzliche Bausteine in diesem Bereich von $A00000 bis $BFFFFF unterzubringen, welche auch gefunden wurde.

CIA-A wird immer dann selektiert, wenn »-VMA« und »A12« Low-Pegel führen. Für die Selektion von CIA-B müssen »-VMA« und »A13« Low-Pegel haben. Dies gilt für den gesamten, oben genannten Speicherbereich. Wenn aber »A12« und »A13« High-Pegel führen, ist kein CIA aktiv. Dies ist die Stelle, wo man eingreifen kann:

Der Baustein 74LS138 ist ein 3-Bit-Binär-Decoder, der die Zustände an seinen Eingangspins 1 bis 3 so decodiert, daß er anhängig vom Zustand dieser Pins einen seiner Ausgänge (Pin 7 und 9 bis 15) auf Low-Pegel geht. Vorausgesetzt, die Pins 4 und 5 liegen auf Low-Signal und Pin 6 führt High-Pegel (die Pins 4, 5 und 6 sind Freigabeleitungen). Der Baustein wurde so verschaltet, daß der Ausgangspin 12 dann Low wird, wenn folgender Zustand an seinen Eingängen herrscht:

A12	High	Pin1
A13	High	Pin2
A20	Low	Pin3
-VMA	Low	Pin4/5

Dies ist immer dann der Fall, wenn eine Adresse mit folgender Maske vom Prozessor ausgegeben wird:

A23 .... A0
1010xxxxxx11xxxxrrrrxxxx

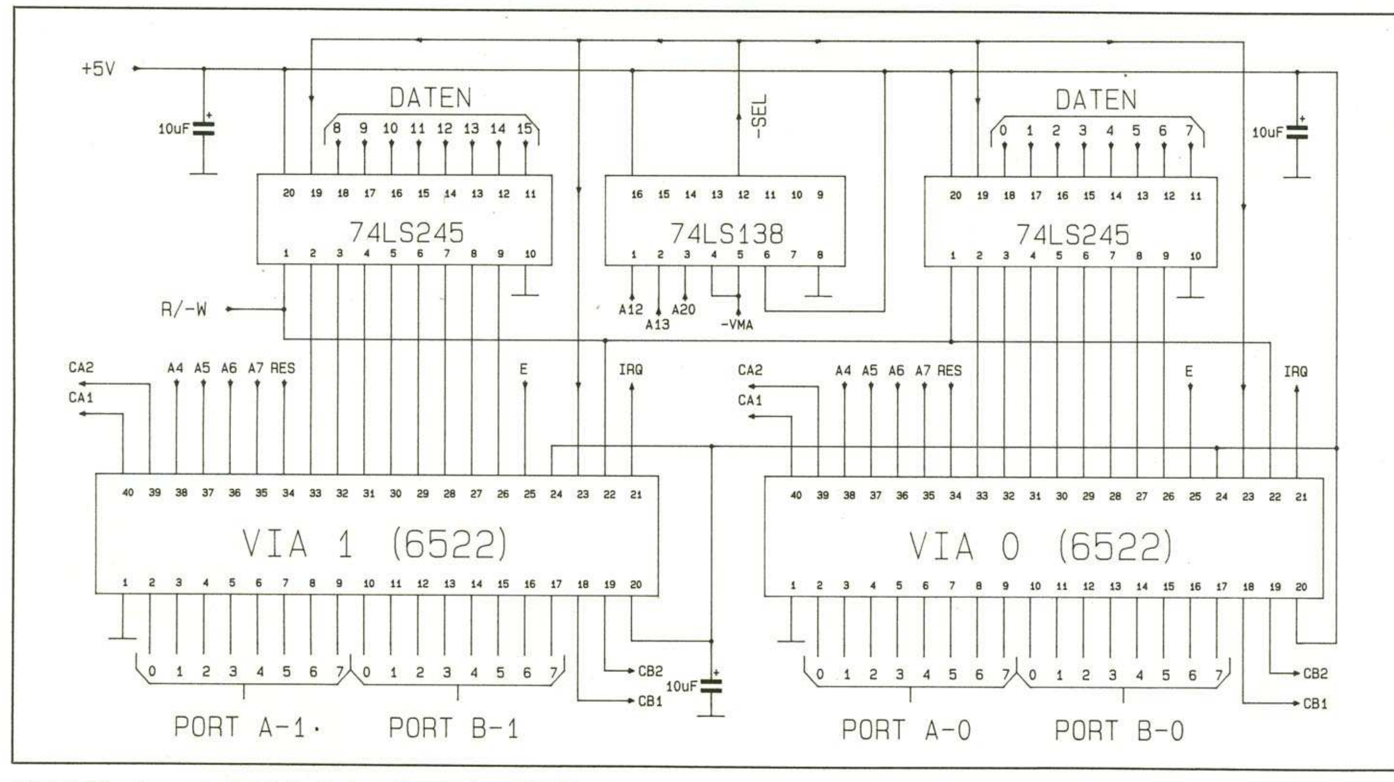


Bild 1. Der komplette Schaltplan der Amiga-VIA-Karte

die Freiheit

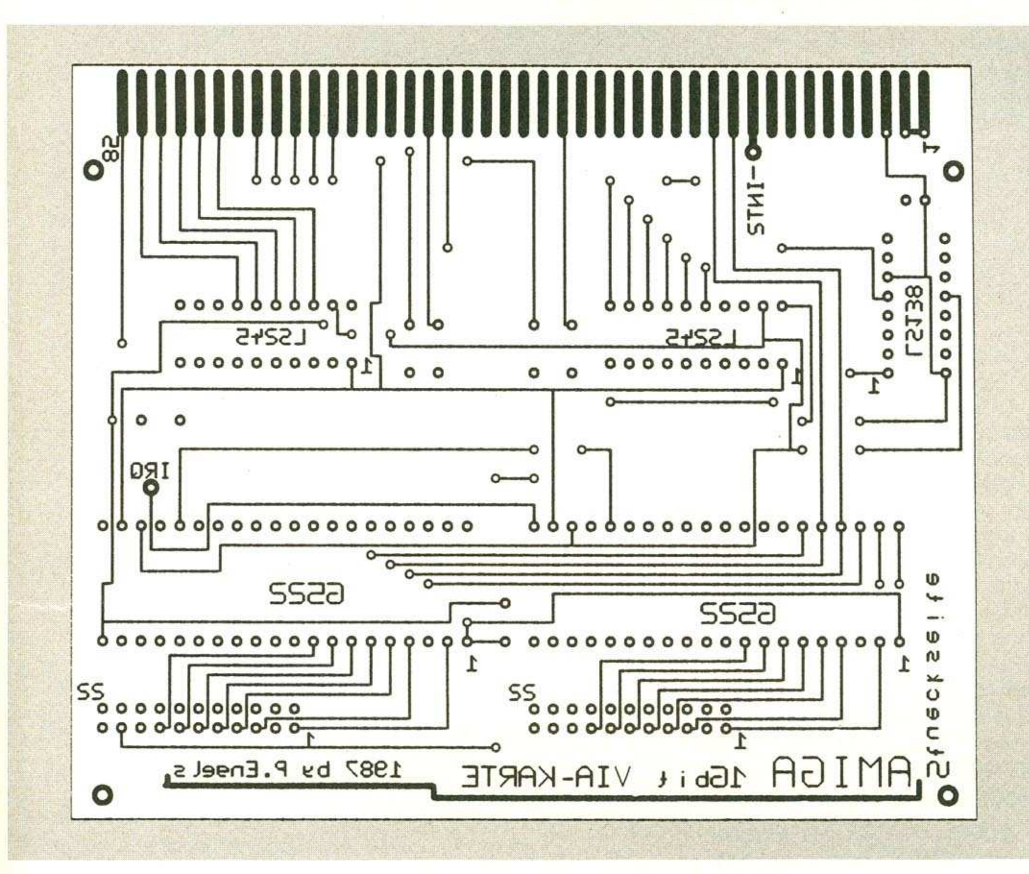


Bild 2. Das Layout für die Bestückungsseite der VIA-Karte im Maßstab 1:1 (spiegelverkehrt)

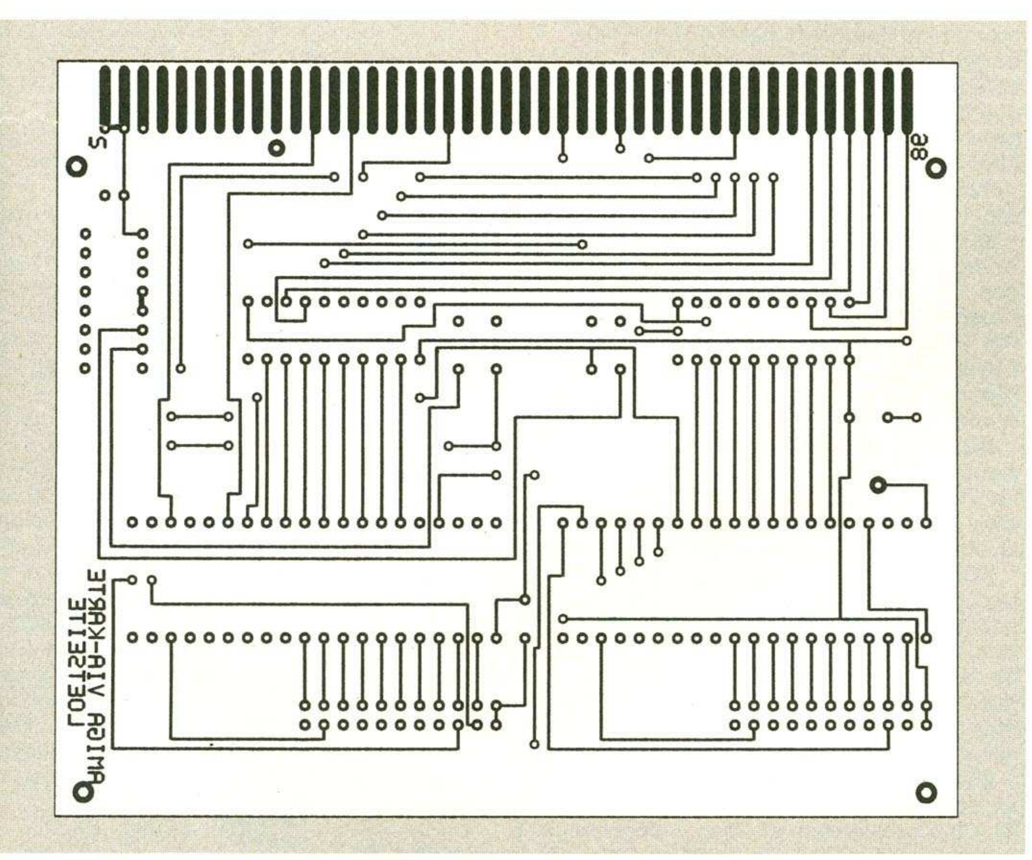


Bild 3. Das Platinenlayout für die Unterseite der VIA-Karte im Maßstab 1:1 (spiegelverkehrt)

welches für folgende Adreßbereiche zutrifft (x = beliebig, r = VIA-Register):
$Ax3000 bis $Ax3FFF
$Ax7000 bis $Ax7FFF
$AxB000 bis $AxBFFF
$AxF000 bis $AxFFFF

Die Startadresse sollte jedoch zwecks einheitlicher Programmierung im Bereich von $A03000 bis $a3F01 für beide VIAs liegen!

Vielfalt und wenig Aufwand

Das so gewonnene Select-Signal »-SEL« wird nun auf den Chip-Select-Eingang der VIAs gelegt, ebenfalls gelangt es an die Enable-Pins der beiden Datenbustreiber 74LS245, die die Datenleitungen puffern und bei Anliegen von »-SEL« auf den Prozessor durchschalten. Die 16 Register der VIAs werden durch die Adressen »A4« bis »A7« ausgewählt. Das bedeutet: die Register folgen sich im Abstand von 16 Byte.

Da VIA0 mit den Datenleitungen »D0« bis »D7« verbunden ist, belegt es die ungeraden Adressen. VIA1, verbunden mit »D8« bis »D15«, belegt die geraden Adressen. Somit werden durch einen Wort-Zugriff wie beispielsweise »MOVE.W $A03000,D« die Inhalte der Ports B beider VIAs in das Prozessorregister »D0« geladen, wobei die unteren 8 Bit den Port-Zustand von VIA0 und die oberen 8 Bit den Port von VIA1 wiedergeben. Somit können durch einen einzigen Wort-Zugriff 16 Output-Pins gesetzt beziehungsweise 16 Inputs gelesen werden. Es ergibt sich hierdurch eine sehr einfache, wortweise Programmierung der Leitungen, die zudem durch den geringen Aufwand auch relativ schnell ist.

Die Ports sind jeweils auf ein 22poliges Rasterfeld gelegt, von wo sie am besten über handelsübliche Stiftleisten mit »dem Rest der Welt« verbunden werden. Es ergibt sich die auf der nächsten Seite folgende Anschlußbelegung:

Pin 1 und 22 sind auf der Oberseite der Platine beschriftet. Es ist folgende Zählweise zu beachten:
2 4 6 8 22
1 3 5 7 21

Pfostenpin	VIA-Pin
1	PA0
2	PB0
3	PA1
4	PB1
5	PA2
6	PB2
7	PA3
8	PB3
9	PA4
10	PB4
11	PA5
12	PB5
13	PA6
14	PB6
15	PA7
16	PB7
17	CA1
18	CA2
19	CB1
20	CB2
21	GND
22	GND

Die doppelseitig zu ätzende Platine ist nach den Platinenlayouts, die in Bild 2 und 3 zu sehen sind, herzustellen. Der Bestückungsplan ist in Bild 4 zu sehen, die einzelnen Bauteile sehen Sie nachstehend:

1 Platine Amiga-VIA-Karte
1 86polige Steckerleiste, 2,54 mm Raster
3 Tantal-Elkos, 10 μF
2 TTL-ICs 74LS245
1 TTL-IC 74LS138
2 VIAs 6522
2 Pfostenleisten 2 x 11polig

Die Bestückung der Platine dürfte keine Probleme bereiten, es ist lediglich auf die richtige Polung der ICs und der Kondensatoren zu achten. Natürlich dürfen auch die vier Drahtbrücken nicht vergessen werden.

Die Platine wird mit den Bauteilen nach oben zeigend in den Expansion-Port des ausgeschalteten Amiga 500 gesteckt (beim A1000 ist zu berücksichtigen, daß der Expansion-Port um 180 Grad gedreht ist), wonach sich der Computer beim Einschalten wie gewohnt verhalten sollte. Ist dies nicht der Fall, untersuchen Sie bitte die Platine auf Lötfehler.

Der Portbaustein 6522 (VIA)

Zum Programmieren der Karte sind Kenntnisse der einzelnen Register des 6522 nötig, deshalb eine Beschreibung der VIAs:

Die VIAs besitzen insgesamt jeweils 16 Register, die die unterschiedlichsten Funktionen erfüllen. Die Tabelle rechts oben erläutert die einzelnen Register und deren Lage im Speicher des Amiga.

Im einzelnen haben die Register folgende Funktionen:

ORB — In diesem Register stellen die einzelnen Bits direkt die entsprechenden Pins »PB0« bis »PB7« dar. Beim Lesen wird der momentane Eingangspegel am Port B übergeben. Falls ein Pin als Ausgang programmiert ist, gibt eine »1« den Ausgang als gesetzt, eine »0« als gelöscht an. Durch Schreiben in dieses Register kann man also einzelne Ausgänge ein- oder ausschalten, vorausgesetzt, der Anschluß ist im DDRB als Ausgang gesetzt. Ist der Port als Eingang programmiert, kann durch Lesen dieses Registers festgestellt werden, welchen logischen Pegel der Eingang hat (0 = Low, 1 = High).

ORA — Es gilt das gleiche wie für ORB, jedoch für den Port A.

DDRB — Hier kann die Datenrichtung eines Pins von Port B festgelegt werden. Eine gesetzte »1« setzt den Pin als Ausgang, eine »0« als Eingang. Nach einem Reset sind alle Pins auf Eingang geschaltet. Um »PB0« bis »PB3« und »PB7« auf Ausgang und alle anderen Pins als Eingang zu programmieren, ist folgender Befehl anzuwenden:

```
MOVE.B #%10001111,$A03020
```

DDRA — Es gilt das gleiche wie für DDRB, jedoch für das Datenrichtungsregister A.

T1C-L, T1C-H, T1L-L, T1L-H — Dies sind die Register für Timer 1. Eine Beschreibung würde den Rahmen dieses Artikels sprengen. Ein Datenblatt mit den Beschreibungen aller Register ist gegen Rückporto (2 Mark) vom Autor erhältlich.

T2C-L, T2C-H — Dies sind die Register für Timer 2.

SR — Diese Zelle kann mit einem Wert beschrieben werden, der, abhängig von der Programmierung, seriell auf Pin »CB2« ausgegeben wird. Ebenfalls können seriell ankommende Daten empfangen werden.

ACR — Das Hilfsregister dient zur Modus-Einstellung für Timer 1 und 2. Ebenfalls kann ein Latching-Modus für die Ports eingestellt werden.

PCR — Dieses Register steuert das Verhalten der Pins »CA2« und »CB2«. Diese Pins können auf an- oder absteigende Flanken reagieren und einen Interrupt auslösen. Diese Pins lassen sich auch als Ausgänge schalten.

IFR — Dieses Register zeigt an, ob ein bestimmter Zustand im Chip aufgetreten ist. Man kann so beispielsweise prüfen, ob ein Timerüberlauf stattfand oder ob an einem flankengesteuerten Input ein Ereignis stattgefunden hat. Die Bits haben folgende Zugehörigkeit:

VIA1	VIA0	Register	Funktion
$A03000	$A03001	ORB	Ausgaberegister Port B
$A03010	$A03011	ORA	Ausgaberegister Port A
$A03020	$A03021	DDRB	Datenrichtungsregister Port B
$A03030	$A03031	DDRA	Datenrichtungsregister Port A
$A03040	$A03041	T1C-L	Timer 1 Low
$A03050	$A03051	T1C-H	Timer 1 High
$A03060	$0A3061	T1L-L	Vorspeicher Timer 1 Low
$A03070	$0A3071	T1L-H	Vorspeicher Timer 1 High
$0A3080	$0A3081	T2C-L	Timer 2 Low
$0A3090	$0A3091	T2C-H	Timer 2 High
$0A30A0	$0A30A1	SR	Schieberegister
$0A30B0	$0A30B1	ACR	Hilfsregister
$0A30C0	$0A30C1	PCR	Steuerregister
$0A30D0	$0A30D1	IFR	Interrupt-Anzeigeregister
$0A30E0	$0A30E1	IER	Interrupt-Enableregister
$0A30F0	$0A30F1	ORA	Ausgaberegister Port A ohne Handshake

Die einzelnen Register der VIAs

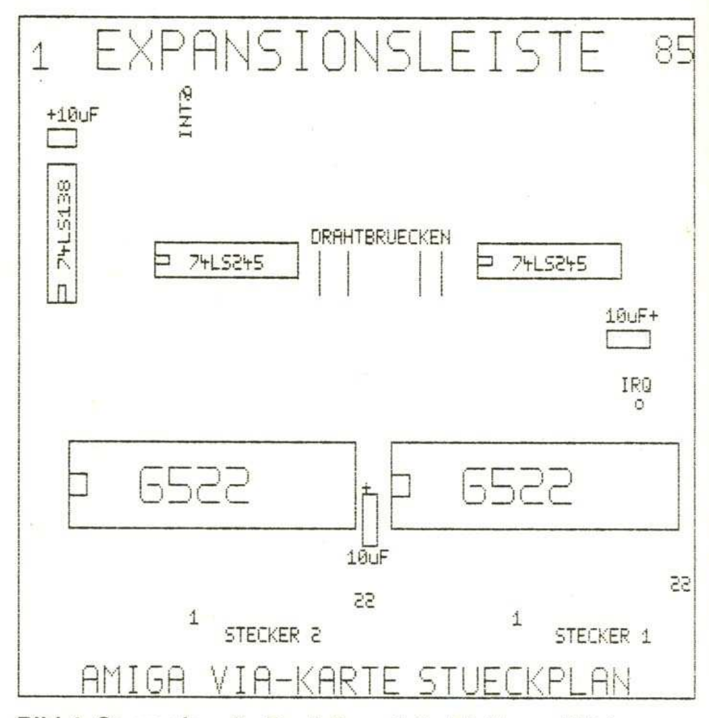


Bild 4. So werden die Bauteile auf die Platine gelötet

Bit:	7	6	5	4
Funktion:		T1	T2	CB1
Bit:	3	2	1	0
Funktion:	CB2	SR	CA1	CA2

Bit 7 ist immer »1«, wenn mindestens ein Bit im IFR »1« ist.

IER — In diesem Register kann bestimmt werden, ob ein gesetztes Bit im IFR ein Low-Signal am IRQ-Ausgang des VIA erzeugen soll. Um allerdings Interrupts verarbeiten zu können, ist der Anschluß IRQ mit »-INT2« mit einem Kabel zu verbinden. Danach können die VIAs ebenfalls einen Interrrupt im Computer auslösen. Die Zugehörigkeit der Bits entspricht denen im IFR.

Dies war eine grobe Beschreibung der VIA-Register. Denjenigen, die weitere Informationen benötigen, sei das Datenblatt über den 6522-VIA von Rockwell International, Doc. No. 29650N4O, empfohlen. Ebenfalls steht der Autor gerne für weitere Fragen zur Verfügung.

Diese Erweiterung dürfte für viele Amiga-Anwender das Tor zur »Hardware-Welt« öffnen. Wir hoffen, daß Ihnen noch viele weitere Anwendungen als die bereits erwähnten mit dieser Karte einfallen, so daß Ihr Amiga noch leistungsfähiger wird, als er jetzt schon ist.

Wer weiß, vielleicht steuern auch Sie bald Ihre Lichtorgel, ein Alarmsystem oder die Zimmerbeleuchtung mit Ihren Amiga? (Peter Engels/dm)

Bei Nachfragen oder Bestellungen wenden Sie sich bitte an folgende Adresse:
Peter Engels
Kreisstr. 29
5308 Rheinbach 14
Telefon: 02226/5714

Wir machen Preise!

HITS DES MONATS

MINDBREAKER	19.–
DEMOLITION	19.–
KING OF CHICAGO	69.–
C-64 EMULATOR	99.–
MOUSE-WASH	19.–
WINTER OLYMPICS 88	59.–
PORTS OF CALL deutsch	89.–
VIDEOTITLER	198.–
PAGE FLIPPER deutsch	85.–
FLUGSIMULATOR II	89.–
GARRISON	59.–
MAXIPLAN PLUS	298.–

ANIMATION

	DM
ANIMATE 3 D	249
ANIMATION TV-Show	219
ANIMATOR & IMAGES AEGIS	269
APPRENTICE DISNEY 3 D JUNIOR	129
APPRENTICE DISNEY 3 D-ANIMATOR	498
APPRENTICE LIBRARIES: GEO	39
APPRENTICE LIBRARIES: LETTERS	39
APPRENTICE LIBRARIES: NUMBERS	39
DELUXE VIDEO II	189
PAGE FLIPPER deutsch	85
SILVER	279
VIDEOSCAPE 3D deutsch	269
VIDEOTITLER AEGIS	198

BÜCHER

	DM
AMIGA JAHRBUCH 1988	12
AMIGA USER'S GUIDE	39
DEVELOPERS REFERENCE GUIDE	49
DOS HANDBUCH AMIGA	15
DOS MANUAL BOOK	59

DATENBANK

	DM
dBman	289
MICRO FICHE FILER	198
SUPERBASE Personal deutsch	219

DEMOS

	DM
DEMO GRAFIKBILDER*1 (IFF)	10

DIVERSES

	DM
C-64 EMULATOR Readysoft	99
DIGA! AEGIS	149
KICK START ELIMINATOR	198

DRUCKER

	DM
CITIZEN 120 D MATRIXDRUCKER	498
FARBBAND CITIZEN 120 D schwarz	14
FARBBAND NEC P2200 schw	19
FARBBAND NEC P6 schwarz	12
FARBBAND NEC P6C color	49
FARBBAND NEC P7 schwarz	15
FARBBAND NEC STAR NL-10 schwarz	12
MPS 1500C FARBMATRIXDRUCKER	798
NEC P6 deutsches Handbuch	1148
NEC P6C deutsches Handbuch	1498
NEC P7C deutsches Handbuch	1898
STAR NL-10C Matrixdrucker	628
Traktor bidirektional NEC P6	349
Traktor bidirektional NEC P7	398
XEROX Diabolo 150 Colorinkjet	1998

FESTPLATTEN

	DM
FESTPLATTE 20 MB FÜR A-2000	1349
FILECARD 20 MB Western Digital	748

GRAFIK

	DM
CALLIGRAPHER	229
DELUXE MAPS GRAFIKBILDER	59
DELUXE PAINT II deutsch	189
DELUXE PRINT & DATA*1 engl.	149
DIGI PAINT PAL deutsch	111
DRAW AEGIS	198
DRAW PLUS AEGIS	429
DYNAMIC CAD	898
FORMS IN FLIGHT 3-D CAD STEREO	149
FUNKTION Graphenzeichner	98
IMAGES AEGIS	69
IMPACT AEGIS	139
INTERCHANGE	99
INTRO CAD	129
PRIXMATE	111
PRINTMASTER PLUS	89
PRISM PLUS	98
SCULPT 3-D RAY-TRACING	180
TV-TEXT 3-D Textanim.	179
X-CAD Techn. Zeichnen	898

GRAFIKDATEN

	DM
ART GALLERY*1	59
ART GALLERY*2	59
ARTPACK-1 AEGIS	59

KABEL

	DM
Adapter Centronics A500/2000	49
Kabel Amiga-Bildschirm 2 mtr.	29
Kabel für Drucker AMIGA 1000	29
NETZKABEL/KALTGERÄTESTECKER	6

KALKULATION

	DM
ANALYZE 2.0 SPREADSHEET	298
LOGISTIX (DEUTSCH)	298
MAXIPLAN 500	229
MAXIPLAN PLUS	329

LAUFWERKE

	DM
LAUFWERK 3 1/2 ZOLL EXTERN	348
LAUFWERK 3 1/2 ZOLL INTERN	328
LAUFWERK 5 1/4 ZOLL EXTERN	448

MONITORE

	DM
MONITOR 1084 (RGB/VIDEO)	648
MONITOR 12" grün TTL	198
MONITOR 14" bernstein TTL	279
MONITOR 14" grün TTL	279
MONITOR 14" weiss TTL	298

MUSIK

	DM
AUDIOMASTER AEGIS	98
DELUXE MUSIC CONSTR. dtsch.	189
DRUM STUDIO	59
DYNAMIC DRUMS	139
FUTURE SOUND II	398
INSTANT MUSIC	85
MUSIC STUDENT	98
MUSIC STUDIO	98
PERFECT SOUND mit Digitizer	198
PRO MIDI STUDIO	298
SONIX MUSIKPROGRAMM, AEGIS	128
SOUND SAMPLER AMIGA 1000	128
SOUND SAMPLER AMIGA 500/2000	128
STUDIO MAGIC	98

SIMULATION

	DM
FLIGHTSIMULATOR II	89
GALILEO PLANETARIUM	98
SCENERY DISK*11	49
SCENERY DISK*7	49
SUPER HUEY	59
SUREGON Operationssimulation	69

SPEICHER

	DM
Speicher 2 MByte AMIGA 2000	839
Speicher 2 MByte AMIGA 500	898
Speicher CBM 512 KByte A-500	275

SPIELE

	DM
A MIND FOREVER VOYAGING	89
ADVENTURE CONSTRUCTION SET	69
ALIEN FIRES adventure	69
ALIEN STRIKE	39
AMEGAS Breakout-Variante	39
AMIGA KARATE Karatespiel	29
ARAZOK'S TOMB Grafikadventure	69
ARCTIFOX 3-D Actiongame	69
ARENA & BRATACASS Sportspiel	79
ARKANOID Breakout	69
BACKGAMMON	34
BACKLASH	49
BAD CATS Katzenolympiade	49
BALANCE OF POWER Grafikadv.	73
BALL RAIDER	29
BALYHOO MYSTERY Adventure	79
BARBARIAN Grafikadventure	59
BARDS TALE Grafikadventure	85
BEAT IT Actiongame	26
BLACK CAULDRONS	69
BORROWED TIME Graficadventure	65
BRAINSTORM	29
BRIDGE Kartenspiel	49
BUREACRACY COMEDY Adventure	79
CHALLENGER Actiongame	26
CHAMPIONSHIP BASEBALL	65
CHAMPIONSCHIP BASKETBALL	65
CHAMPIONSHIP FOOTBALL	77
CHAMPIONSIP GOLF	65
CHESSMASTER 2000 3D-Schach	75
CLEVER & SMART Actionadv.	54
COMPUTER BASEBALL	68
COOGANS RUN	49
CRAZY CARS	69
DARK CASTLE Action	55
DEFENDER OF THE CROWN	69
DEJA VU Grafikadventure	79
DEMOLITION Action	19
DESTROYER Schlachtschiffsim.	69
DETONATOR Superbreakout	39
DIABOLO	39
DR. FRUIT Action	26
EARL WEAVER BASEBALL	95
ENCHANTER Adventure	59
FAERY TALE Grafikadventure	69
FUED	29
FIGHTER COMMAND Action	89
FINAL TRIP Action	26
FIREPOWER Action	49
FLIP FLOP Brettspiel	26
FOOTMAN	59
GALACTIC INVASION Action	49
GALAXY FIGHT Action	29
GARRISON II Action	59
GARRISON Strategie/Action	59
GOLDEN OLDIES div. Spiele	59
GOLDEN PATH	49
GOLDEN PYRAMID Strategie	39
GOLDRUNNER Action	73
GRAND SLAM TENNIS	79
GRIDIRION Foorball	119
GUILD OF THIEVES Grafikadv.	79
HACKER II Adventure	75
HALLEY PROJECT Grafikadv.	69
HEX GRAPHIC/STRATEGY	89
HITCHHIKERS GUIDE Adventure	59
HOLLYWOOD HIJINX Adventure	79
HOLLYWOOD POKER Strip-Poker	45
HUNT FOR RED OKTOBER	69
IMPACT Breakout	49
INSANITY FIGHT	59
INTO THE EAGLES NEST Action	65
JEWELS OF DARKNESS Adventure	69
JIGSAW MANIA	29
JINGXTER Action	69
KAMPFGRUPPE Strategie	98
KARATE KID II Action	65
KARATE KING Action	35
KARTING GRAND PRIX	29
KICKSTART II Motocross	29
KING OF CHICAGO (PAL)	69
KINGS QUEST 3 Grafikadv.	89
KINGS QUEST I+II+III	99
KNIGHT ORC Grafikadv.	75
KWASIMODO	29
LAS VEGAS	29
LEADER BOARD GOLF	65
LEADER BOARD TOURNAMENT	39
LEATHER GODESSES Adventure	79
LEVIATHAN	49
LITTLE COMPUTER PEOPLE	69
LURKING HORROR	79
LYBIANS IN SPACE Actionspiel	59
MAGICIANS DUNGEONS Grafikadv.	79
MARBLE MADNESS Murmelspiel	63
MERCENARY COMPENDIUM	69
MIND FOREVER Adventure	79
MINDBREAKER Action	19
MISSION ELEVATOR Actionadv.	49
MOEBIUS	69
MOONMIST Adventure	79
MOUSETRAP	44
NINJA MISSION	29
NORD AND BERT	79
OGRE Action	79
PACK BOY Action	25
PHALANX Action	25
PHANTASIE III WRATH OF NIKAD	98
PLUDERED HEARTS	79
PLUTOS Action	59
POKER Kartenspiel	79
PORTAL Grafikadv.	89
PORTS OF CALL deutsch	89
Q-BALL	49
QUINTETTE Strategie	85
ROADWAR 2000 Straßenkrieg	98
ROCKY	29
S.D.I.	79
SANTA PARAVIA	69
SHADOWGATE	79
SHANGHAI	69
SHOOTING STAR	26
SILICON DREAMS	69
SINDBAD & THRONE O. FALCON	69
SPACE BALLER	19
SPACE BATTLE	25
SPACE FIGHT	29
SPACE QUEST	98
SPACE RANGER	29
SPACEPORT	49
SPEED car racing	25
STARFLEET I	89
STARGLIDER game	79
STARWARS Weltraumabenteuer	59
STATIONFALL	79
STOCK MARKET	69
STRANGE NEW WORLD	45
STREETGANG Banden in New York	59
STRIP POKER	69
TASS TIME IN TONETOWN	79
TEMPLE OF ASPHAI	68
TENTH FRAME	79
TERRAMEX	55
TERRORPODS	59
TESTDRIVE	65
TETRIS	49
THAI BOXING	29
THE ART OF CHESS	69
THE PAWN adventure	79
TIME BANDITS Action	49
TRINITY	79
ULTIMA III	69
UNINVITED	69
VADER	29
VIDEO VEGAS	59
VYPER	59
WINTER GAMES	59
WINTER OLYMPICS 88	49
WISHBRINGER	39
WIZARDS CROWN	99
WORLD GAMES	59
XENON Action	59
XR-35	29
ZORK Trilogy(I+II+III)	119

SPRACHEN

	DM
AC BASIC Compiler	349
AZTEC C COMMERCIAL	849
ATZEC C DEVELOPERS	479
AZTEC C PERSONAL	398
LATTICE C COMPILER V 4.0	448
LISP METACOMCO	498
MACRO ASSEMBLER METACOMCO	169
MODULA-2 COMMERCIAL	598
MODULA-2 DEVELOPER	298
MODULA-2 REGULAR	198
PASCAL (ISO) METACOMCO	220
PASCAL USCD	298

SYSTEME

	DM
AMIGA 2000 GRUNDGERÄT	2295
AMIGA 500 GRUNDGERÄT	1098

TEXT

	DM
CITY DESK	249
DESKTOP ARTIST	79
FAST FONTS	79
FLOW V1.02 IDEA PROC.	149
PAGESETTER deutsch	289
PAGESETTER LASERSCRIPT	98
PRO WRITE	239
PROFESSIONAL PAGE	598
SCRIBBLE	198
VIZAWRITE Desktop deutsch	189
VIZAWRITE DESKTOP engl.	189
ZUMA FONTS Vol. 1	59
ZUMA FONTS Vol. 2	59
ZUMA FONTS Vol. 3	59

TOOLS

	DM
BUTCHER GRAPHIC-CONV.+EDITOR	69
CLI MATE	64
DEMONSTRATOR deutsch	89
DISCovery DISK EDITOR deutsch	198
DISK TO DISK	98
DOS EXPRESS	59
DOS TO DOS	98
FLOPPY ACCELERATOR	59
GIZMOZ 2.0	128
GRABBIT	59
MARAUDER	59
MIRROR	98
MIRROR HACKER PACKAGE	98
POWER WINDOWS C'Libs & Help	149
PROJECT D	79
SHELL METACOMCO	98
TOOLKIT METACOMCO	89
TxEd Editor europ. Version	79
ZING KEYS	89
ZING!	139

VIDEO

	DM
DIGI VIEW DIGITIZER dtsch.	269

ZUBEHÖR

	DM
AT-ERWEITERUNG FÜR AMIGA 2000	1695
BASIC KEYBOARD OVERLAY	19
DISKETTEN 3 1/2 ZOLL 2 DD	2,35
DISKETTEN 5 1/4 2DD (farbig)	10
DISKMAPPE 3 1/2 ZOLL	23
DOS KEYBOARD OVERLAY	19
EASYL 1000 ZEICHENTABLETT	798
EASYL 2000 ZEICHENTABLETT	848
EASYL 500 ZEICHENTABLETT	698
JITTER-RID FILTERSCHEIBE	29
JOYSTICK Competition Pro	25
JOYSTICK JOYSTAR (Metall)	19
Kabel CENTRONICS PC & AMIGA	15
MIDI INTERFACE A-1000	98
MIDI INTERFACE A-500/2000	98
MONITORSTÄNDER	39
MOUSE PAD (9 1/2" x 11")	18
PAL VIDEO KARTE AMIGA 2000	149
TV Modulator für AMIGA	59
XT-Erweiterung für AMIGA 2000	1098

Preisänderungen und Irrtümer vorbehalten

amigaland
a. koppisch
hohenwaldstraße 26
6374 steinbach

Bestellservice:
06171/71846

Alle Preise sind Endpreise.
Mindestbestellwert DM 50,-.
Versand ausschließlich gegen Nachnahme.

Die ... der Fantasie

Interessieren Sie sich für Rollenspiele; haben sich aber bisher noch an keines herangetraut? Hier erhalten Sie die wichtigsten Informationen, um den Kampf in fremden Dimensionen heil zu überstehen.



Die wichtigsten Elemente in den meisten Rollenspielen sind: Kämpfer, Monster, fremde Land

Die ursprünglichste Form der Rollenspiele ist ein Gesellschaftsspiel, das sich einige Studenten amerikanischer Universitäten zum Zeitvertreib ausgedacht haben. Mehrere Personen treffen sich dabei zu einer Sitzung mit einem Dungeon-Master als Vorsitzender. Der Master ist der erfahrenste Spieler und muß alle Regeln kennen. Er überwacht das gesamte Spielgeschehen und übernimmt die Steuerung der Monster. Die übrigen Spieler bilden nun eine Gruppe (Party), die sich durch eine imaginäre Welt voller Abenteuer und Überraschungen, aber auch voller Gefahren schlagen muß. Die Welt in der dies alles geschieht, existiert also ausschließlich in den Köpfen der Spieler und ist durch in Büchern enthaltene Beschreibungen sowie ein umfangreiches Regelwerk bis ins letzte Detail charakterisiert. Jedes Spiel hat außer einer umfangreichen Hintergrundgeschichte natürlich auch eine Aufgabe, die es zu erfüllen gilt. Bevor man jedoch einen Schatz finden, einen bösen Zauberer besiegen oder eine Welt retten kann, muß man erst einmal viele einzelne Aufgaben lösen und Erfahrungen sammeln.

Da es relativ schwierig ist, ständig alle Regeln im Kopf zu behalten und außerdem noch eine geeignete Anzahl von Mitspielern zu finden, hat man schon relativ früh damit begonnen, solche Spiele auf Computern umzusetzen. Reihen wie Ultima, Phantasie und Bard's Tale machen inzwischen auf vielen verschiedenen Computertypen Furore. Der Computer übernimmt dabei die Rolle des Dungeon-Master, während es nur noch einen Spieler gibt, der aber dann gleich eine ganze Gruppe von Figuren durch das Spiel steuert. Damit entfällt natürlich eine Menge Arbeit und man kann sich ganz auf das Abenteuer selbst konzentrieren. Nebenbei wird auf dem Bildschirm noch eine Grafik gezeigt und meist ist auch ein bißchen Sound integriert.

Der Bildschirm eines Rollenspiels ist deswegen in mehrere Bereiche aufgeteilt, aus denen man die verschiedenen Informationen ersehen kann.

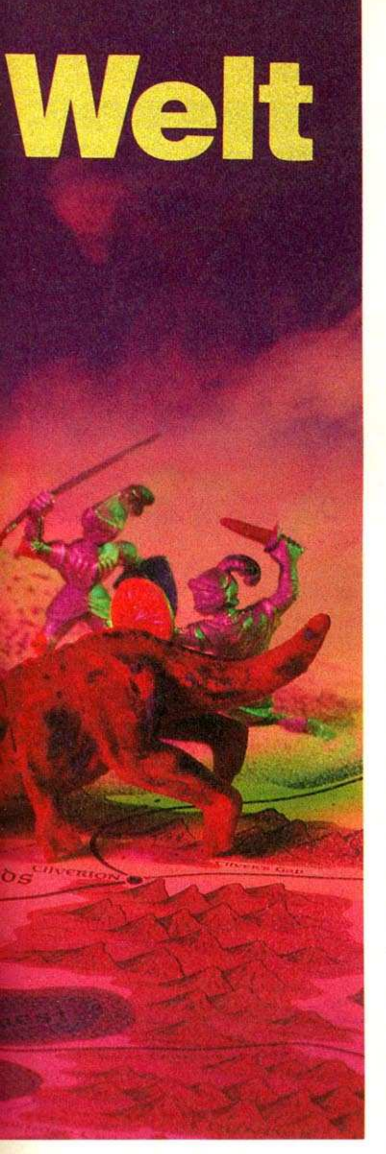
Viel auf dem Bildschirm

Ein großer Teil wird meist von einem Grafikfenster eingenommen. Darin wird die Umgebung, in der sich die Gruppe der Abenteurer gerade befindet, dargestellt. Dies wird bei den verschiedenen Rollenspielen durchaus auf unterschiedliche Weise bewerkstelligt. Eine sehr verbreitete Methode ist die Darstellung einer Landkarte mit entsprechenden Markierungen. Diese Art Karte findet man zum Beispiel vorwiegend in Ultima oder Phantasie. Der Standort der Party wird gesondert vermerkt, während bestimmte Symbole für die verschiedenen Geländeformationen stehen (Bild 1). Wenn die Karte so gut entworfen ist wie in »Phantasie III«, sieht man die einzelnen Felder, aus denen sie besteht, nicht mehr. Eine flüssige Bewegung der Gruppe über die Karte ist jedoch nicht möglich. Darum wird auch kein Joystick als Steuerung benutzt, sondern die Cursor-Tasten. Das Symbol für die Spieler springt immer von einem Feld auf das nächste.

Eine andere Methode der Darstellung verwendet beispielsweise »The Bard's Tale«. Dieses Programm zeigt keine Landkarten, sondern entwirft eine dreidimensionale Ansicht der momentanen Umgebung aus der Sicht der Spielergruppe (Bild 2). Es bleibt einem selbst überlassen, eine detaillierte Karte zu zeichnen, denn eine grobe Übersicht wird dem Programmpaket beigelegt. Bei Bard's Tale ist die Grafik durchgehend in dieser Art ausgestaltet. Es gibt jedoch auch Mischformen, da zusätzlich jedes Rollenspiel noch Höhlenlabyrinthe die sogenannten Dungeons enthält. In Ultima werden diese unterirdischen Gewölbe als 3D-Labyrinth gezeichnet, während ansonsten Karten zu sehen sind. Phantasie hält sich wiederum ausschließlich an die Draufsicht (Bild 3).

Das Konzept dieser Dungeons läßt sich folgendermaßen erklären: Die Gesamtaufgabe eines Rollenspiels ist in mehrere Teilaufgaben zerlegt. Ähnlich einem Puzzle muß man sich die einzelnen Teile für die Lösung zusammensuchen. Die Dungeons beherbergen meist wichtige Anhaltspunkte oder Gegenstände, die zur Lösung der Gesamtaufgabe beitragen. Jeder Gang und alle Räume dieser Verliese müssen aufs genaueste abgesucht werden. Das Problem ist nur, daß in den Dungeons meist viel gefährlichere Monster anzutreffen sind als im offenen Land. Viele Dungeons kann man nur durchsuchen und bezwingen, wenn die einzelnen Spielfiguren der Gruppe stark genug sind und besondere Fähigkeiten erlernt haben.

Daraus läßt sich ersehen, daß dem Aufbau der Spielfigu-



schaften und der Zufall

ren sehr viel Zeit und Sorgfalt gewidmet werden muß. Diese Figuren gehören zu unterschiedlichen Rassen, haben verschiedene Berufe und damit verbundene Eigenschaften und sind körperlich sowie geistig in variabler Verfassung.

Sind Sie charakterstark?

An Rassen werden Menschen, Zwerge, Elfen, Gnome, Halblinge und manchmal auch verschiedene monsterähnliche Kreaturen angeboten. Zu Beginn eines Abenteuers muß man sich eine Gruppe von Figuren aus den vorhandenen Rassen aussuchen. Auch die mögliche Klasse, seltener Beruf genannt, darf man noch bestimmen. Den Rest der Charaktereigenschaften bestimmt der Computer per Zufallszahl und man muß sich damit zufrieden geben, oder die Prozedur des Erschaffens einer Figur von vorne wiederholen.

Zur interessantesten Klasse in jedem Fantasy-Rollenspiel gehören natürlich die Zauberer oder ähnliche magisch begabte Personen. Hier schwanken die Bezeichnungen am häufigsten und gehen von Wizard über Magican bis zu Sorcerer. Eine Gruppe Abenteurer kann ohne sie genausowenig auskommen, wie ohne kräftige Kämpfer, die sich furchtlos den Monstern in den Weg stellen. Unter diesen gibt es Klassen wie Warrior, Fighter oder Paladin. Ein weiterer klassischer Beruf der Rollenspiele ist der des Diebes. In vielen Spielen braucht man einfach einen Dieb, um an bestimmte Schätze ohne viel Aufwand heranzukommen. Manchmal kann nur er vertrackte Schlösser an wichtigen Türen knacken. Die wichtigsten Rassen sind damit zwar schon abgedeckt, um jedoch das Spiel richtig interessant werden zu lassen, sind noch einige Mischklassen entworfen worden. So sind zum Beispiel Priester und Mönche durchaus in der Lage auch bestimmte magische Sprüche zu lernen und können trotzdem mit der Waffe ihren Mann stehen. Die meisten richtigen Zauberer haben mit Waffen und Rüstungen nicht viel am Hut. Jäger und Ranger wiederum vereinen teilweise die Vorzüge von allen drei Basisklassen in sich, haben aber wie alle anderen auch ganz spezielle Nachteile. Die richtige Mischung in einer Gruppe kann nicht über den Sieg, aber über das schnelle Vorankommen der Party entscheiden.

Der Computer ermittelt allerdings nicht nur mit Zufallszahlen die geistigen und körperlichen Eigenschaften der Figuren, sondern läßt eine Kombination von Rasse und Klasse mit in die Berechnungen einfließen. Zwerge und Menschen geben sehr gute Kämpfer ab. Daher haben sie schon einen gewissen Bonus bei der Verteilung der dafür besonders nötigen Eigenschaften wie Stärke und Kampfkraft. Aus kleinen Rassen wie etwa Gnomen oder Halblingen werden eher hervorragende Diebe.

Die Attribute, die jede Figur des Spiels ausmachen, werden in Zahlenwerten angegeben. Die häufigsten individuellen Attribute sind Stärke, Intelligenz, Geschicklichkeit, Konstitution und Charisma.



Bild 2. Eine Stadt aus dreidimensionaler Sicht

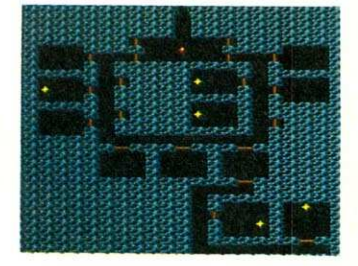


Bild 4. Die Karte eines Dungeon mit ihren Kammern

Stärke: Die Stärke eines Charakters gibt nicht nur an wie wirksam er im Kampf zuschlagen kann, sondern ist auch wichtig für die Menge an Waffen und Rüstung, die er mit sich herumschleppt.

Intelligenz: Intelligente Personen können als einzige die vielen Zaubersprüche verwenden. Außerdem lernen solche Figuren eher und schneller neue Sprüche dazu.

Geschicklichkeit: Je geschickter jemand ist, um so einfacher kann er einem Angriff oder einem gegen ihn gerichteten Zauberspruch ausweichen. Andererseits erhöht sich dadurch auch seine Wirksamkeit im Kampf, da er die Waffen besser führen kann.

Konstitution: Dies bezeichnet im wesentlichen die physische Verfassung einer Person und wird meist direkt unter der Bezeichnung »Hit Points« (Trefferpunkte) ausgedrückt. Je mehr Hit Points vorhanden sind, um so mehr Treffer kann man einstecken bevor der Tod eintritt.

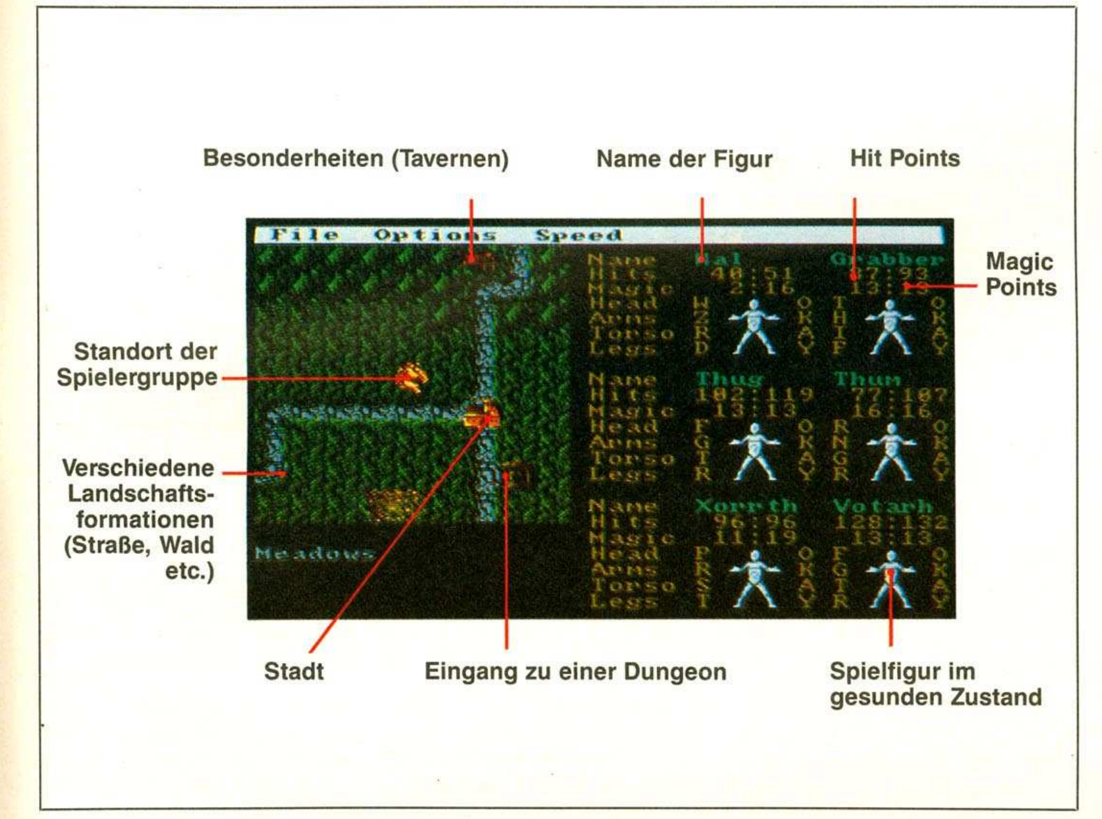


Bild 1. Der Bildschirm eines typischen Rollenspiels ist in verschiedene Felder unterteilt

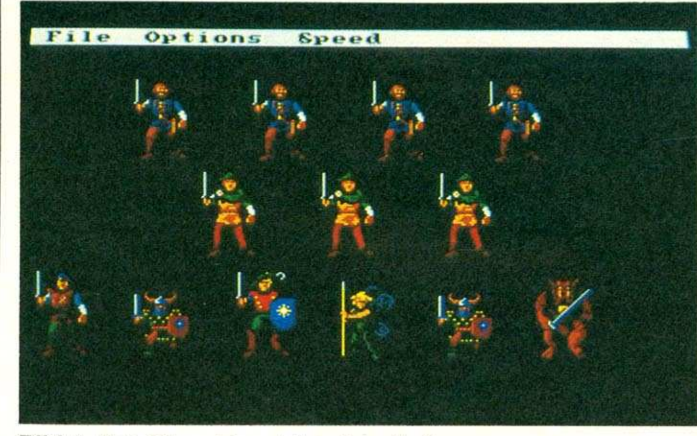


Bild 4. Ihre Charaktere kämpfen als Gruppe

Charisma: Charaktere mit hohem Charisma profitieren davon auf den Gebieten Diebesglück, Zauberkraft, Verhandlungsgeschick und Lernen (Training) neuer Fähigkeiten.

Nachdem die Figuren zur Zufriedenheit entworfen und eine gute Gruppe zusammengestellt worden ist, kann das Spiel begonnen werden. Zunächst sind die Charaktere jedoch noch auf dem untersten Rang und haben weder sehr viel Kampfkraft noch besonders viele Zaubersprüche. Erste Aufgabe ist demnach, so viel Erfahrung und Gold zu sammeln, um in den nächsten Level aufzusteigen. Dies ist meistens eine der schwierigsten und frustrierendsten Phasen des ganzen Spiels. Es ist sehr gefährlich, sich weit von einer Stadt zu entfernen, oder gar tief in einen Dungeon zu gehen. Einzelne oder kleinere Gruppen von Monstern sollte man sich vornehmen (Bild 4). Nach einem Sieg bleibt meist ein Geldbetrag zurück und außerdem erhält jede Figur die wichtigen Erfahrungspunkte (Experience Points). Immer wenn eine bestimmte Menge dieser Punkte gesammelt wurde, ist eine Aufwertung des Charakters möglich. Ein höherer Rang ist für eine Figur mit vielen Verbesserungen der Charaktereigenschaften verbunden. Oft hat man mehr Kraft und Hit Points als vorher oder bekommt die Chance neue Zaubersprüche zu lernen. Damit kann sich die Gruppe dann noch stärkeren Gegnern stellen und in unbekannte Gebiete vorstoßen. Lernen, trainieren und vor allen Dingen den Spielstand speichern, kann man nur in den sicheren Städten oder wie bei Bard's Tale im Haus der Gilde der Abenteurer. Auch zum Aufstocken eventuell benötigter Lebensmittel müssen Städte aufgesucht werden. Wer nach einiger Zeit genug Gold gesammelt hat, um sich bessere Waffen und eine verläßlichere Rüstung zuzulegen, darf dies natürlich in den Waffenschmieden tun. Besondere Häuser einer Stadt sind dann noch die Tavernen oder Wirtschaften. Darin findet man außer Erholung und einem Glas Wein auch meist einen abgewrackten, aber redseligen alten Kämpfer, der einem für ein bißchen Gold einen wertvollen Tip gibt.

Zauberkundig

Sollten die Figuren einigermaßen aufgebaut sein, so daß man überall herumlaufen kann ohne Angst zu haben, die nächste Stadt nicht mehr zu erreichen, werden nach und nach die Gruppenmitglieder mit magischen Fähigkeiten immer wichtiger. Die Zaubersprüche lassen sich nicht nur zum Kämpfen benutzen, obwohl auch einige unter ihnen sind, die eine ganze Gruppe von gefährlichen Angreifern in die Flucht schlagen. Von großer Wichtigkeit sind vor allem die Sprüche zur Heilung von Wunden und Giften. Solche heilende Magie findet ihre Krönung in der Wiederbelebung im Kampf gestorbener Mitkämpfer. Andere Anwendungen der Magie sind beispielsweise die Teleportation der gesamten Gruppe an einen anderen Ort. Sogar durch Wände oder Decken kann der geübte Zauberkundige gehen und Fallen im Nu erkennen und von Ferne unschädlich machen.

In Rollenspielen auf dem Computer liegt die besondere Faszination einer Welt mit eigenen Regeln, die es zu erforschen und zu erobern gilt. Wer ein bißchen Geduld und Zeit aufbringen kann, sollte sich diese Art von Spielen nicht entgehen lassen. Wann spielen Sie Ihre nächste Rolle? (jk)

Das Schloß der Schatten

AMIGA test

Fertigmachen zum Besuch in gefährlichen, verrückten, seltsamen aber auch witzigen Welten zweier neuer Abenteuerspiele. »Jinxter« und »Shadowgate« bieten Unterhaltung par excellence. Sowohl für Einsteiger als auch für Profis ist etwas dabei; wir sagen Ihnen welches das Richtige für Sie ist.



...oder die Sache mit

Wenn Softwarefirmen Nachfolger zu bereits erfolgreichen Spielen herausbringen, sind das nicht immer nur Plagiate. In seltenen Fällen kann die hervorragende Qualität der Vorgänger gehalten, ja sogar noch gesteigert werden. Nicht nur Adventurefreaks sind die beiden Grafikabenteuer »Deja Vu« und »Uninvited« positiv aufgefallen (siehe auch AMIGA 1/88). Die Programmierer von »Icom Simulations« haben sich durch eine einfallsreiche und für Adventure neue Benutzeroberfläche hervorgetan. Shadowgate heißt der neueste Streich dieser bei »Mindscape« erschienenen Reihe, die sich nicht nur an eingefleischte Abenteurer wendet.

Auch von »Magnetic Scrolls« gibt es mit Jinxter wieder ein Adventure mit hervorragender Grafik (Bild 1). Der Handlungsort ist diesmal allerdings nicht wie in den Vorgängern »The Pawn« und »Guild of Thieves« das Land Kerovnia, obwohl sich viele Anlehnungen und Anspielungen finden.

Hier geht es vielmehr um das Glück des Landes Aquitania und dies kann man sehr wörtlich nehmen. Um die böse, schwarze Magie zu bannen, hat dort vor Jahren der Zauberer Turani einen Talismann entworfen, der mit fünf Glücksbringern bestückt ist. Solange dieser Talismann existiert, filtert er die schwarze Magie und läßt nur die ungefährliche grüne Magie in Aquitania bestehen. Damit kann man allerlei Streiche veranstalten, doch niemals anderen etwas wirklich Böses antun. Außerdem haben alle Bewohner des Landes unendlich viel Glück und können sich nicht ernsthaft verletzen. Der Frieden wird jedoch nachhaltig duch die garstige Hexe Jannedor gestört, die nach dem Ableben des Zauberers Turani nach der Macht trachtet. Sie will die schwarze Magie wiederbeleben und hat einige uneinsichtige Bewohner dazu überredet, die Glücksanhänger vom Armband zu entfernen und zu verstecken. So nach und nach verliert sich die Kraft des Armbandes und das Glück schwindet aus Aquitania. Hier setzt die Handlung von Jinxter ein, denn Sie sollen versuchen alle fünf fehlenden Anhänger (Charms) wiederzufinden und damit die Hexe zu besiegen.

Was sind eigentlich Guardians

Zur Seite steht Ihnen dabei das seltsame Völkchen der Wächter (Guardians), die eine fast gottgleiche Kaste von Überwesen sind. Sie hat der Zauberer Turani vor seinem Tod aus anderen Dimensionen gerufen, um die Leute des Landes zu beschützen. Doch das Problem mit den Wächtern ist, sie haben zwar gottgleiche Fähigkeiten und sind unsterblich, doch ansonsten gleichen sie vergeßlichen und verbohrten kleinen Stammtisch-Philosophen oder Beamten. Sie laufen ständig in altenglischer Kleidung mit obligatorischem Fischgrätenmuster herum, essen mit Vorliebe den stinkigen Cheddar-Käse und benutzen eine Sprache, die voll ist von Slang- und Beamtenausdrücken.

Dies spiegelt sich natürlich auch in den Texten des Adventures wider. Jinxter ist zwar ein Grafikadventure, doch die etwa 30 Bilder zu verschiedenen Räumen sind reine Stimmungsmacher. Sie wurden diesmal von verschiedenen Künstlern gezeichnet, was sich in durchaus unterschiedlichen Qualitäten bemerkbar macht. Kaum ein Bild ist übrigens so seltsam wie das der ersten Szene aus dem Bus. Die restlichen haben die gleiche Qualität, wie etwa in »Guild of Thieves«. Doch worauf es im wesentlichen ankommt ist der Text. Wer nicht ein sehr festes Standbein in der englischen Sprache hat, wird bei Jinxter echte Probleme bekommen. Durch die seltsame Sprache der Guardians ist schon die Anleitung zum Spiel äußerst schwierig zu verstehen. Vom Konzept her gehört sie jedoch zum Besten, was es auf diesem Gebiet gibt. Sie ist in Form einer Tageszeitung aus Aquitania gestaltet und beinhaltet in den verschiedenen Artikeln einige Tips und Hintergründe zum Spiel. So wird gleich auf der ersten Seite deutlich, wie die fünf Glücksbringer aussehen, die es zu finden gilt: Einhorn, Walroß, Pelikan, Drache und Feuerwehrauto.

Aber nicht nur die Anleitung und die sprachlich sehr pfiffigen, aber eben auch sehr schwierigen Texte machen das Abenteuer zu einem Spiel für echte Adventure-Profis. Die Probleme, die sich stellen, sind diesmal von allererster Güte.

Das Glück in Aquitania

Da man in Aquitania zu Beginn des Spiels eine wirklich wahnsinnige Menge an Glück hat, kann einem vorerst einmal nicht das Geringste passieren. Jedesmal wenn sich eine tödliche Falle auftut, kommen die Wächter zu Hilfe und bereinigen die Situation. So kommt man praktisch im Spiel immer weiter, doch der Schein trügt. Durch »Score« kann man abfragen wieviel Glück man noch besitzt und die zu Beginn erhaltene Menge darf, um das Spiel zu gewinnen, nicht um ein Quentchen abnehmen. Da ist zum Beispiel die Sache mit dem Bullen auf der Weide (Bild 2). Man kommt nicht an ihm vorbei, um in den nächsten Raum zu gelangen. Die einzi-



dem Glück

ge (einfache) Möglichkeit, die zu sehen ist, stellt ein rotes Tischtuch dar. Schwenkt man es, so kommt der Bulle natürlich sofort auf einen zugedonnert und würde einen zu Tode trampeln, wenn... ja wenn da nicht die Wächter wären. Kurz bevor der Koloß aus Muskeln und Hörnern Sie niederwalzt, tut sich vor Ihnen eine Grube im Boden auf, in der einige Guardians geschäftig am Buddeln sind. Da der Bulle nicht mehr stoppen kann, fällt er in die Grube und Sie sind ihn für immer los. Doch diese Lösung hat ihr Glück arg strapaziert, Sie fühlen sich gleich nicht mehr so wahnsinnig glücklich, sondern eben nur noch super glücklich. Auf die richtige Lösung zu kommen heißt in diesem Fall das rote Tuch zusätzlich weit von sich zu werfen, noch bevor die Wächter eingreifen. Der Bulle wird dann auf das Tuch und nicht auf Sie losgehen, und Sie haben die Situation gemeistert, ohne Glück zu verlieren.

Das Adventure ist gespickt mit den verzwicktesten Puzzlen dieser Art und kann daher nicht gerade für Einsteiger empfohlen werden. Der von Magnetic Scrolls gewohnte sehr gute Parser, der mit einem umfangreichen Wortschatz glänzt und auch verzwickte Sätze mit vielen Vokabeln versteht, macht jedoch dem etwas geübteren Spieler das Leben zur Freude. Überhaupt kann das gesamte Konzept voll überzeugen und sprüht nur so von guten Einfällen und witzigen Ausführungen. Man kann dem Programmierer-Team nur einen weiteren wirklich gelungenen Wurf bescheinigen.

Für wen das aber alles zu vertrackt ist, und wer sich nicht mit dem Eintippen von Texten herumschlagen möchte, der kann jetzt mit eingangs angesprochenem Shadowgate trotzdem auf Adventure-Pfaden wandeln. Möglich wird dies durch das ausgefeilte Bedienungssystem, das seine Programmierer dort verwirklicht haben. In Bild 3 sieht man den gesamten Bildschirm des Spiels, der mit seinen Fenstern dem Aussehen der Workbench sehr ähnlich sieht. Die Windows lassen sich genauso verschieben oder in der Größe verändern, wie man es gewohnt ist. Spezielle Fenster wie etwa die Grafik (links oben) oder die Befehle (rechts oben) sind natürlich immer von der gleichen Größe. Als sehr wichtiges Fenster steht in der Mitte die »Inventory«, in dem sich alle Gegenstände befinden, die Sie im Moment bei sich tragen.

Adventure mit Fenstern

Text findet sich bei Shadowgate nur im unteren Window und beschränkt sich ausschließlich auf Kommentare und Beschreibungen. Die Steuerung des Spiels geschieht vollkommen mit der Maus. Dies macht Shadowgate besonders für Einsteiger interessant, da man nur einfache, verständliche Texte lesen muß und der Rest dem Arbeiten auf der Workbench gleicht. Will man zum Beispiel einen Gegenstand aufnehmen, so braucht man ihn nur in der Grafik mit der Maus anzuklicken und in das Inventory-Fenster

hinüberzuziehen. Alle Besonderheiten in einer Grafik, die sich bewegen, öffnen oder in sonst irgendeiner Weise manipulieren lassen, verändern wie ein Icon beim Anklicken ihre Farbe. So kann man schnell herausfinden, welche Gegenstände interessant sind. Das Öffnen von Türen funktioniert beispielsweise auf drei verschiedenen Methoden. Die erste Möglichkeit wäre, die Tür in der Grafik einmal anzuklicken und dann »OPEN« aus dem Befehlsmenü auszuwählen. Anders geht es auch mit dem »Exits«-Window. In diesem Fenster werden alle sichtbaren Ausgänge aus einer jeweiligen Szene als kleines rotes Kästchen dargestellt. Ein Klick auf das zu der Tür gehörende Kästchen mit anschließendem OPEN wäre also auch denkbar. deswegen »OPERATE«. Das Konzept ist einfach erklärt am Vorgang des Aufschließens einer Tür verdeutlicht. Zuerst wird der passende Schlüssel angeklickt, dann geht man auf OPERATE und als letztes wählt man die Tür. Wenn es der richtige Schlüssel war, bekommt man die Meldung »The Door is now unlocked«. Als kleiner Tip gilt, daß OPERATE auch von einem Gegenstand auf denselben gelenkt werden kann. Will man beispielsweise eine Fackel in der Fassung drehen, um einen Geheimgang freizulegen, so geht dies nur durch ein OPERATE von der Fackel auf die Fackel.

Die Thematik, um die es bei Shadowgate geht, liegt diesmal voll im sogenannten Fantasy-Bereich. Schade ist nur, daß man in der recht gut gemach- konnte ihm und seinen Heerscharen von üblen Kreaturen kaum etwas entgegensetzen, so daß sich schließlich der Zirkel der Zwölf entschloß einzugreifen. Talimar wurde mit aller Kraft aus dem Land verbannt und von da an nur noch der »Warlock Lord« genannt.

Die Prophezeiung erfüllt sich

Doch als nach vielen Jahrhunderten, nachdem auch das letzte Mitglied des Zirkels dahingegangen ist, die Macht des dunklen Lords wieder spürbar wird, soll sich eine alte Prophezeiung erfüllen. Einer aus dem Zirkel wird zurückkehren und dafür sorgen, daß der Warlock Lord vernichtet wird und wieder ein rechtmäßiger König über das Land herrscht. die jeder Beschreibung spotten. Machen Sie sich gefaßt auf goldgierige Drachen, gemeine Trolle, gefräßige Höllenhunde und andere gruselige Erscheinungen (Bild 4). Obwohl das Schloß eine Vielzahl von Räumen besitzt, sind nur wenige für Normalsterbliche zu erreichen. Trotzdem müssen Sie versuchen den Warlock Lord daran zu hindern, die Welt mit Dunkelheit zu überziehen. Sein Ziel ist es nämlich aus den Tiefen der Erde unterhalb Shadowgate den letzten und tödlichsten aller Titanen zum Leben zu erwecken, den »Behemoth« (Bild 5).

Es gibt eine Menge zu tun, doch die Rätsel haben fast immer eine sehr logische Lösung. Man sollte jedoch darauf achten, die Grafik einer Szene aufs genaueste mit dem Maus-



Bild 3. Komfortables Spielen mit vielen Fenstern auf der hervorragenden Benutzeroberfläche: »Shadowgate«



TITEL	Shadowgate
Spielidee / Grafik / Sound / Schwierigkeit / Motivation	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
Besonderheiten	Komfortables Adventure mit Sound
Hersteller	Mindscape
Preis	77 Mark
Bezugsquelle	Soyka Datentechnik, Hattinger Str. 685, 4630 Bochum 5

Bild 4. Der »Firedrake« ist nur eines von vielen Monstern. Haben Sie den »Crystal Ball« noch?

Bild 5. Dies ist kein Kaffeeplausch. Können Sie den Aufstieg des Behemoth verhindern?

Die einfachste Methode bleibt aber ein schlichter Doppelklick auf die Tür oder das Kästchen. Das Exits-Window ist übrigens dazu da, um auch außerhalb des Blickwinkels liegende Ausgänge oder später entdeckte Geheimtüren anzuzeigen.

Viele Methoden führen zum Öffnen

Wenn Sie zum Beispiel ein Behältnis finden, das man öffnen kann, so wird bei entsprechender Wahl dieses Menüpunktes ein Extra-Fenster geöffnet. Dies zeigt den Inhalt etwa eines Beutels an und man kann nun andere Sachen in diesen Sack hineinlegen oder herausholen und danach den Beutel (vielmehr sein Fenster) wieder schließen. Die Bedienung wird zum absoluten Kinderspiel, da es noch einen Menüpunkt gibt, der alle schwierigen Operationen an irgendwelchen Sachen übernimmt und er heißt auch wahrscheinlich ten Anleitung viel über die Art zu spielen erklärt bekommt, jedoch relativ wenig über die Hintergrund-Story ausgesagt wird. Deshalb hier exklusiv ein paar Informationen für alle, die noch nicht so tief in die Geheimnisse des Schlosses Shadowgate eingedrungen sind.

Frieden herrschte vor langer Zeit im Land Tarkus, als die Mitglieder des »Zirkels der Zwölf« noch ungehindert ihren Forschungen nachgehen konnten. Dieser Verbund der zwölf erfahrensten Druiden hat seit den Anfängen der Zeit die Könige beraten und das Land beschützt. Doch es gab unter ihnen einen, den man als Talimar den Schwarzen bezeichnete, und er forschte am liebsten im Bereich der schwarzen Künste. Nachdem man seine üblen Neigungen erkannt hatte, konnte der Zirkel ihn nicht länger dulden. Er wurde ausgestoßen und sann auf Rache. Nachdem er genug Macht angesammelt hatte, versuchte er den König zu stürzen. Dieser

So geschieht es zu Beginn des Adventures. Sie werden als letzter Nachfahre der Königslinie von dem Druiden Lakmir dem Zeitlosen vor einen Eingang in die Katakomben des Castle Shadowgate teleportiert. Dies war einst der Stammsitz der Druiden und jetzt treibt der böse Lord dort mit Kreaturen sein Umwesen,

zeiger abzusuchen. Nur so können besondere Gegenstände oder eventuell Geheimtüren gefunden werden.

Um die Stimmung noch zusätzlich zu erhöhen, wurden in Shadowgate einige Soundeffekte eingebaut. Noch öfter als in den beiden Vorgänger-Abenteuern haben die Programmierer auf diese Effekte zurückgegriffen. Da gibt es die obligatorischen knarrenden Türen oder das gefährliche Fauchen eines Drachen. Richtige Schauer jagt einem aber erst das hohle Lachen des Warlock Lord über den Rücken, das sich an den Wänden der endlosen Gänge bricht.

Mit seiner leichten Bedienung, hübschen Grafiken und überraschenden Sounds macht Shadowgate sowieso schon recht viel Spaß beim Spielen. Aber auch mit der Story und den spannenden Aufgaben braucht es sich nicht hinter Jinxter zu verstecken und ist sein Geld allemal wert. Wer jedoch auf der Suche nach richtig schwierigen Adventure-Aufgaben ist und eine Beschäftigung über lange Zeit sucht, der kommt um Jinxter nicht herum.

(jk)



TEXTOMAT AMIGA

Egal, ob Sie Briefe oder ein ganzes Buch schreiben wollen, TEXTOMAT AMIGA einlegen und losschreiben – ohne langes Anlernen und Lesen im Handbuch. Mit Maus und Menü haben Sie den enormen Leistungsumfang von TEXTOMAT AMIGA schnell im Griff: alle Textverarbeitungsfunktionen, vielfältige Blockoperationen (Kopieren, Verschieben, etc.), hohe Geschwindigkeit bei der Ein- und Ausgabe von Texten, Direktformatierung am Bildschirm (WYSIWYG), Grafikeinbindung (IFF-Format), Kopieren und Einlesen von Bildschirmausschnitten aus anderen Programmen, automatische Silbentrennung, bis zu 30 Funktionstasten mit bis zu 160 Zeichen belegbar (als Floskeltasten, Mini-Adressdatei oder Tastaturmakros), beliebig viele Text- und Dezimaltabulatoren, Datentransfer über RS 232, Lesen von ASCII- und Notepad-Dateien, Datenaustausch mit anderen Programmen über Clipboard, fertige Druckeranpassungen für alle gängigen Drucker, 2 verschiedene Zeichensätze für Monitor- und Fernsehbetrieb, ausführliches deutsches Handbuch. TEXTOMAT AMIGA – die Textverarbeitung für alle zum nahezu konkurrenzlosen Preis.

TEXTOMAT AMIGA **nur DM 99,–**

DATAMAT AMIGA

Die Dateiverwaltung, die auch Bilder und Grafiken verwalten kann: DATAMAT AMIGA. Rundum professionelle Features: Programmsteuerung über Maus und Tastatur, frei gestaltbare Bildschirmmaske, komfortable Such- und Selektierfunktionen, Paßwortschutz, Datenaustausch mit anderen Programmen, Einbindung von Grafiken im IFF-Format – DATAMAT AMIGA kann alles, was man von einer Dateiverwaltung für den AMIGA erwartet. Weitere Pluspunkte in Kürze: Dateigröße max. 2 Milliarden Zeichen, max. 8 offene Dateien gleichzeitig, Datensatzgröße max. 64000 Zeichen, max. 2 Milliarden Datensätze, unbegrenzte Anzahl der Datenfelder, max. Feldgröße 32000 Zeichen, max. 20 Indexfelder mit wählbarer Genauigkeit (1-99) Zeichen, Suchen und Selektieren auch nach Bereichen und Und-/Oder-Verknüpfungen, Bildschirmmaskengenerator unterstützt Grafikelemente wie Rechteck, Kreis, Linie, Muster, verschiedene Textarten und -größen, mehrzeilige Textfelder mit Wortumbruch- und Formatierungs-Möglichkeiten, integrierter Druckmasken- und Listen-Editor, ausführliches deutsches Handbuch. Der besondere Clou dieser pfiffigen Dateiverwaltung ist ihr günstiger Preis.

DATAMAT AMIGA **nur DM 99,–**

BECKERbase AMIGA

Der professionelle Datenbank-Manager zum Kaum-zu-glauben-Preis arbeitet nach dem Netzwerkmodell. Diese Struktur gestattet die Definition komplexer Dateiverbindungen mit schnellem Datenzugriff. Für eigene Anwendungen kann eine komfortable Benutzeroberfläche mit Pull-Down-Menüs und Window-Technik programmiert werden. Zwei leicht beherrschbare Programmiersprachen (DDL und TDL) garantieren hohe Flexibilität in der Anpassung an spezielle Benutzerwünsche. Mehrere fertige Demo-Anwendungen sind bereits installiert (Adreß-, Artikel-, Kunden-, Literaturverwaltung u. a. m.). Datenaustausch mit anderen Programmen ist möglich (ASCII-Format). Integrierter Text-Editor, Hilfe-System, einfache Datei-Definition, praktisch unbegrenzte Anzahl von Datensätzen je Datenbank, 65535 Datensätze pro Datei, praktisch unbegrenzte Anzahl von Feldern je Datei, max. Feldgröße 255 Zeichen, Paßwortschutz, ausführliches deutsches Handbuch, kurz: BECKERbase Amiga hat alles, was man für ernsthafte Datenbank-Anwendungen braucht. Minimalkonfiguration: 1 MByte RAM und Kickstart 1.2

BECKERbase AMIGA **nur DM 99,–**

PROFIMAT AMIGA

Das komplette Programmentwicklungspaket für alle Nutzer, die ihren Amiga über die Maschinensprache voll ausreizen wollen. Mit allen Features, die der engagierte Programmierer erwartet: Integriertes Programmsystem bestehend aus Editor, Debugger, Disassembler und Reassembler, läuft unter CLI und Workbench, extrem schnell, da in Assembler geschrieben, mit umfangreicher Betriebssystembibliothek, MAKROS mit beliebig vielen Parametern unterschiedlichen Typs möglich, volle 32-Bit-Arithmetik, Fehlersuchfunktion, Cross-Referenzliste, bedingte und wiederholte Assemblierung menügesteuert, Debugger mit 68020 Single Step Emulation, erzeugt optional PC relativen und absoluten Code. PROFIMAT AMIGA – das vielseitige Programmentwicklungspaket für den Amiga zum sensationellen Preis läuft auf jedem Amiga mit 512 KByte und Kickstart 1.2

PROFIMAT AMIGA **nur DM 99,–**

COUPON

HIERMIT BESTELLE ICH

NAME, VORNAME

STRASSE, ORT

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Die Schlacht um Scandor

»Phantasie III« erfordert vom Spieler taktische Überlegungen, stellt viele Rätsel und ist unterhaltend bis zum Schluß — dem entscheidenden Kampf mit Nikademus.

In Scandor herrscht eine trügerische Ruhe. Die Bevölkerung in den Städten bemerkt noch nicht die Bedrohung, die über dem Land schwebt. Doch im Süden der Insel toben bereits schwere Kämpfe. Der mächtige Nikademus hat einen Pakt mit den Göttern abgeschlossen: Auf der Ebene des Lichts haben Pluto und Zeus ihm erlaubt, eine Burg zu errichten. Von dieser Festung des Grauens dringt die Brut des Nikademus in die Welt der Sterblichen. Niemand ist in der Lage, den Eroberer zu stoppen — niemand? Doch — ein tapferer Held wie Sie kann »Phantasie III« lösen. Alles was Sie brauchen ist Mut und ein Amiga mit 512 KByte. Bei größerem Speicher muß »NoFastMem« in der »Startup-Sequence« eingebunden werden. Auf dem A1000 müssen Sie zusätzlich das zweite Laufwerk ausschalten. Ach ja, Sie brauchen auch das englische Handbuch, um die ständigen Fragen zu beantworten, die das Programm im Spielverlauf stellt. Mit dieser Methode möchte der Hersteller verhindern, daß »Phantasie III« Raubkopierern zum Opfer fällt.

Flinke Finger

In der Stadt Pendragon beginnt das Abenteuer. Rotten Sie eine Gruppe von sechs Mutigen zusammen: Sie haben die Wahl zwischen Kämpfern, Dieben, Priestern und Zauberern. Jeder besitzt ganz spezielle Eigenschaften, die auf der vor Ihnen liegenden Reise wichtig sind: Kraft, Intelligenz, Gewandheit, Kondition und Charisma. Zusätzlich besitzen die Abenteurer noch unterschiedlich ausgebildete Fähigkeiten in mehreren Disziplinen:

— Kämpfer schützen die Gruppe durch Ihre Stärke.
— Die Diebe sind flink, finden Fallen und besitzen die Gabe, Schlüssel zu organisieren.
— Priester heilen.
— Zauberer vermögen Kraft ihrer Magie die schrecklichsten Feinde zu besiegen. Sie führen den entscheidenden Schlag gegen Nikademus.

Doch bis zum Finale ist es ein weiter Weg. In vielen Kämpfen muß die Gruppe Erfahrung und Gold gewinnen. Bei jeder Begegnung mit den Kreaturen von Scandor haben Sie mehrere Optionen:

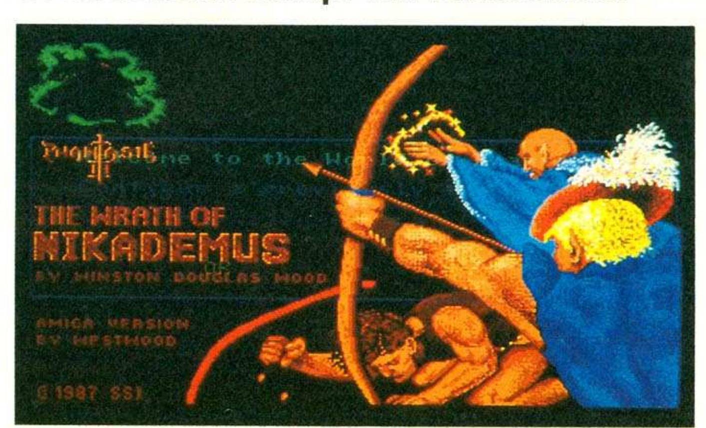
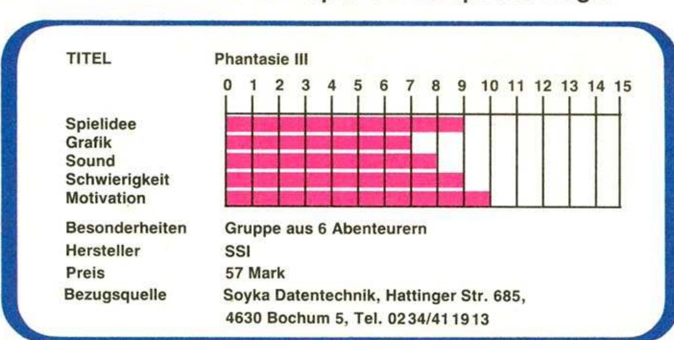


Bild 1. Phantasie III ist ein Spiel um Kampf und Magie



TITEL	Phantasie III
Besonderheiten	Gruppe aus 6 Abenteurern
Hersteller	SSI
Preis	57 Mark
Bezugsquelle	Soyka Datentechnik, Hattinger Str. 685, 4630 Bochum 5, Tel. 0234/411913

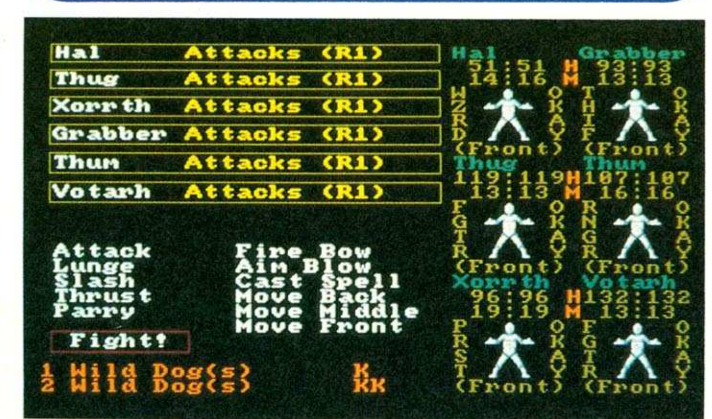


Bild 2. Die richtige Wahl der Waffen ist wichtig

— Flehen Sie um Gnade,
— Wenn Sie fliehen, verlieren Sie Hab und Gut.
— Am besten, Sie greifen an. Mit der Maus oder über die Tastatur wählen Sie für jeden Ihrer Helden einen Kampfstil (Bild 2): Alle Beteiligten schwingen ihre Schwerter, schießen Pfeile oder schleudern magische Sprüche, jedoch läßt die Animation etwas zu wünschen übrig. Ein Gemetzel endet mit der Aufgabe oder Vernichtung einer Partei.

Gewinnt Ihre Gruppe, bekommt jeder Streiter »Experience«-Punkte. Erreicht die Zahl eine bestimmte Höhe und besitzen Sie genügend Gold, können Sie in der nächsten Stadt drei Fähigkeiten eines Kämpfers »trainieren«.

Die Steigerung der Schlagkräftigkeit und das Erlernen effektvoller Zaubersprüche ist ein Ziel von »Phantasie III«. Das zweite ist es, einen Weg zur Festung des Nikademus zu finden. Hierzu müssen die Suchenden in Höhlen hinabsteigen, mächtige Monster besiegen, magische Waffen finden und zahlreiche Rätsel lösen. Anhaltspunkte finden Sie auf zahlreichen »Scrolls«. Die ersten Tips holen Sie jedoch im Archiv von Pendragon ein. Der weise Filmon stellt Ihnen dort eine Aufgabe. Wenn Sie den Auftrag erfüllt haben, erhalten Sie bei Ihrer Rückkehr den nächsten. Bis Filmon Sie dorthin schickt, wo sich das Schicksal von Scandor erfüllt.

Dunkler Dämon

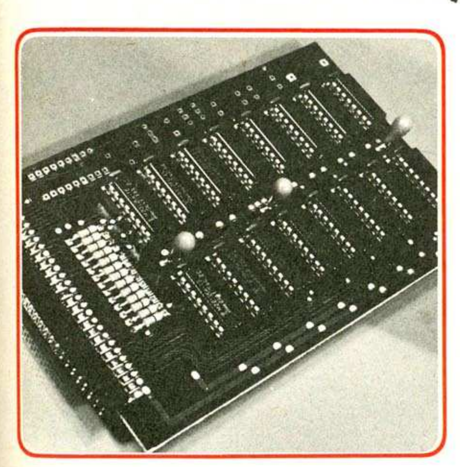
An die Weisungen von Filmon sollten Sie sich halten. Verzetteln Sie sich nicht gleich zu Beginn, indem Sie sich an die härtesten Brocken heranwagen: Keiner der sechs unerfahrenen Recken würde den Kampf gegen einen »King-Dragon« überleben. Am Anfang heißt es, kleine Brötchen zu backen — sprich hier und da eine Spinne oder einen Zwerg besiegen — und schnell zurück in die Stadt. Ein Nachteil: Wenn die durchtrainierten Abenteurer sich dem Ziel des Spiels nähern, stellen sich auch noch die zuvor gefürchteten Gegner in den Weg. Die häufigen Kämpfe machen das Spiel langatmig.

In den Städten tauschen Sie die erbeuteten Waffen, Zaubertränke und Heiltinkturen ein oder verteilen Sie auf die Gruppe. Vergessen Sie in der Stadt nicht, dem Waffenladen und der Dorfkneipe einen Besuch abzustatten. Dort erlangen die Geschwächten ihre volle Stärke zurück.

Ein guter Trick, um den Verlust der Kämpfer zu vermeiden: Speichern Sie in jeder Stadt den Spielstand ab. Wenn ein unerwartet starker Feind den unerschrockenen Helden den Garaus macht, beginnen Sie einfach mit dem zuletzt gespeicherten Spielstand. Mehr Tips wollen wir Ihnen aber nicht geben, sonst verliert »Phantasie III« seinen Reiz. Bliebe noch zu erwähnen, daß Sie nach der Erfüllung des letzten Auftrags zwar von den Göttern geehrt werden, aber nach Scandor zurückkehren. Speichern Sie dort zum letzten Mal die Daten Ihrer siegreichen Mannen. Sie werden Sie brauchen, wenn sich der Besiegte erneut — mächtiger denn je — über dem Land erhebt. (ub)

ausgereifte Ingenieurleistung ● 14 Tage Umtauschrecht ● 2 Jahre Garantie ● fast alle IC'S gesokkelt ● nur professionelle Leiterplatten ● Bauteile namhafter Hersteller ● mit Bedienungsanleitung ● Blockschaltbild ● teilweise Schaltplan





500er Speichererweiterung

Für 512k zusätzliches RAM ● Alle RAM's gesokkelt ● Selbstkonfiguzierend ● Abschaltbar ● Uhrenschaltung auf Platine mit Akku- bzw. Batteriepufferung nachrüstbar

Komplett mit 512K	Preis auf Anfrage
Superpreis mit Uhr	Preis auf Anfrage
Bauteilesatz für Uhr ohne Akku	24.-
Leerplatine mit Stecker	*39.-

* mit Schaltplan und Bestückungsliste



Profilaufwerk 3,5"

● Metallgehäuse ● einstellbare Laufwerknummer mit Displayanzeige ● Digitale Trackanzeige ● Write Protect am Laufwerk schaltbar ● abschaltbar ● durchgeschleifter Bus

349.-

TEAC FD 135 FN 3,5"

1MB ● 1 Zoll (2,54 cm) hoch 239.-

3,5" Laufwerk

Für alle Amiga's ● einstellbare Gerätenummer ● Abschaltbar ● Metallgehäuse ● Superflach 1 Zoll (2,54 cm) ● TEAC Lauwerk

Komplett anschlußfertig 298.-

Laufwerk 5,25"

● 40/80 Track ● Laufwerkbus durchgeschleift ● abschaltbar ● einstellbare Adressen 339.-

HD 1,6 MB (umschaltbar) 359.-

Gemischtes Doppel 3,5"/5,25"

● einzeln ein-/abschltbar ● einstellbare Laufwerksnummern mit Anzeige ● durchgeschleifter Bus ● bei 5,25" 40/80 Tracks umschaltbar ● Metallgehäuse

598.-

Trackanzeige

● für alle Laufwerke (3,5"/5,25") ● Laufwerkbus-durchgeschleift ● mit Gehäuse

49.-

Laufwerkanschlußkabel

Zum Anschluß von Laufwerken an alle Amiga's Mit Ansteuerelektronik 39.-

3-fach Steckplatzerweiterung für Laufwerke

Jeder Steckplatz abschaltbar und einstellbare Laufwerknummer ● Steckplatzerweiterung direkt am Amigagehäuse ● Dadurch keine Kabellängenpropleme

Anschlußfertig zum Alcompsuperpreis von 49.-



Soundsampler

Für Amiga 1000 und 500 mit Software ● Type bei Bestellung bitte angeben ● 8-Bit Datenbreite ● Betrieb am Paralellport (Druckerport) ● Mit Vorverstärker für Micro-Anschluß (Cinch-Buchsen) ● Musik- und Sprachdigitalisierung möglich ● Arbeitet mit fast allen Digitizer-Programmen ● Formschönes Gehäuse

Superpreis 79.-

MIDI-Interface

4 Kanäle einschließlich 1 Thru ● Optische Datenanzeige ● Formschönes Gehäuse

Wahnsinnspreis von nur 89.-

Bootselector 19.90

Kickstartumschaltung

● Bauen Sie die anderen Kickstart-Versionen in Ihren Amiga 500 ● Einfacher Einbau ohne Löten ● Für Original-KickstartROM und 2 zusätzliche Versionen auf EPROM

59.-

Bestellung und Versand

ALCOMP
A. Lanfermann
Lessing Str. 46
5012 Bedburg
Tel. 0 22 72/15 80

Nachnahmeversand NN-Spesen 7.50 DM b. Vorkasse 3.- DM. Auslandsbestellungen: Nachnahmeversand NN-Spesen 10.- DM b. Vorkasse 5.-DM. Wir liefern Ihnen auf Ihre Rechnung und Gefahr zu den Verkaufs- und Lieferbedingungen des Elektronikgewerbes. Postgiroamt Köln (BLZ 370 100 50) 275 54-509

Wir suchen ständig Hardware-Entwicklungen. Wir garantieren gute Umsatzprovisionen und ehrliche Abrechnung

TITEL M

AMIGA test

Eine der wichtigsten Anwendungen des Amiga beim »Desktop Video« ist das Erzeugen von Titeln für attraktivere Filmvorspänne. Der »Video Titler« verbindet jetzt Texte aller Größen und Formen in abwechslungsreichen Animationen.

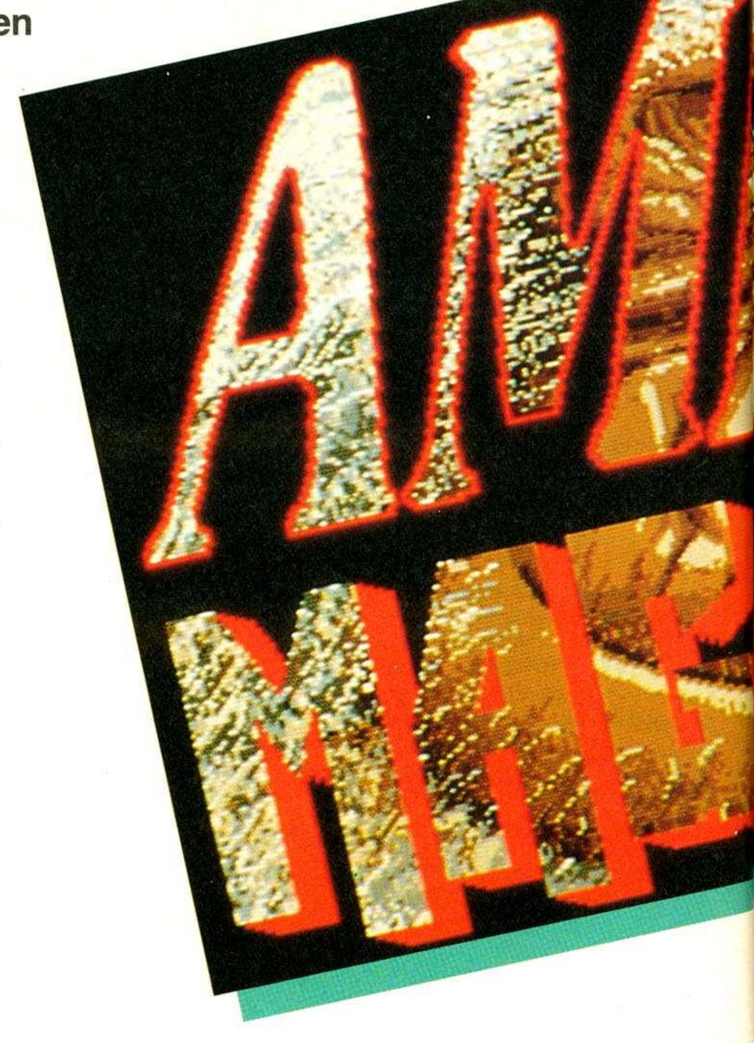
Grob gesagt handelt es sich beim »Video-Titler« um ein Programm, mit dem das Schreiben von Text in verschiedenen Zeichensätzen auf den Bildschirm so leicht wie möglich gemacht werden soll. Der Bildschirmhintergrund läßt sich mit einem oder mehreren IFF-Bildern »bekleben«, wobei die Bilder auch noch verzerrt und gekippt werden können. Um die Ergebnisse auf Video zu verbessern, kann mit dem Overscan-Modus der Bildschirmrahmen abgeschaltet werden. Sogar der Extra-Halfbrite-Modus wird unterstützt. Auch an die europäischen Amiga-Besitzer wurde gedacht: Die PAL-Auflösung wird vom Video-Titler bereits verwendet. Um mit dem Video Titler überhaupt arbeiten zu können, benötigt man übrigens unbedingt zwei Diskettenlaufwerke und mindestens 1 MByte Speicher. Sollen Titel in den Auflösungen 640 x 256 und 640 x 512 erstellt werden, so sind sogar 1,5 MByte notwendig. Normalerweise wird man aber in der 320 x 200-Punkte-Auflösung arbeiten.

Vielseitige Zeichensätze

Eigentlich besteht der Video-Titler aus zwei Programmen: dem Titelprogramm selbst und dem »Video Seg«, was soviel wie »Video-Spezial-Effekt-Generator« bedeutet. Mit dem Video-Titler-Programm werden die eigentlichen Titel erstellt und auf Diskette gespeichert, während »Video Seg« dafür zuständig ist, die gespeicherten Bilder hintereinander abzuspielen. Dabei stehen verschiedene Methoden zum Überblenden der Bilder zur Verfügung.

Der erste Schritt beim Produzieren eines Video-Titels ist natürlich die Eingabe der Texte, die auf dem Bildschirm erscheinen sollen. Jede Textzeile wird vom Video-Titler als spezieller Text betrachtet, der eine eigene Farbe und einen eigenen Zeichensatz haben kann. Natürlich lassen sich auch mehrere Textstücke nebeneinandersetzen. Man ist also bei weitem nicht auf eine Farbe oder einen Zeichensatz pro Zeile beschränkt. Der Hauptvorteil gegenüber einem Grafikprogramm mit Textfunktion ist, daß auch bereits geschriebene Texte jederzeit angeklickt und mit den Cursor-Tasten und Delete verändert werden können. Außerdem lassen sie sich jederzeit per Mausklick umpositionieren. Leider geschieht dies nur zeilenweise, was etwas unpraktisch ist, wenn man etwa zehn Textzeilen nach unten rücken will.

Um einen Text auf den Bildschirm zu bringen, wird erst einmal ein Zeichensatz benötigt. Es werden die normalen Amiga-Zeichensätze verwendet, von denen über zwanzig mitgeliefert werden. Natürlich können auch andere von Diskette geladen werden. Zusätzlich lassen sich auch die »ColorFonts« des Fonteditors »Calligrapher« einsetzen. Bis zu zehn Amiga/Colorfonts können gleichzeitig im Speicher stehen. Sie werden mit einer Funktionstaste aufgerufen. Sind die gewünschten Fonts einmal im Speicher, so wird die Einstellung auf Diskette gespeichert. Die gewünschten Zeichensätze werden dann beim Laden des Programmes mitgeladen. Amiga- und ColorFonts können vom Programm fett, kursiv oder unterstrichen dargestellt werden. Bei einigen Fonts führt der Versuch, sie kursiv zu schreiben, allerdings zum Totalabsturz des Programmes.



Die Amiga/Colorfonts kann man nicht in der Größe verändern oder verzerren. Das würde die Schriftqualität zu stark beeinträchtigen. Hier geht das Programm einen anderen Weg: mit Hilfe der von Aegis entwickelten »PolyFonts« lassen sich Zeichen in beliebiger Größe auf den Bildschirm bringen. Die PolyFonts sind auf der Diskette als Polygonzüge gespeichert, also als eine Folge von Zeichenbefehlen, die das Programm beim Schreiben der Zeichen ausführt. Diese Zeichen lassen sich beliebig drehen, verzerren, vergrößern, kursiv schreiben und verkleinern. Die Abstände zwischen ihnen sind einstellbar, außerdem lassen sie sich entweder gefüllt oder nur als Außenlinie darstellen. Zusätzlich zu den fünf auf Diskette mitgelieferten PolyFonts gibt es noch einen »Symbol«-PolyFont, mit dem sich zum Beispiel Rahmen leicht auf den Schirm bringen lassen. Leider haben die PolyFonts einige Nachteile: Den fünf mitgelieferten fehlen teilweise Zahlen und sonstige Sonderzeichen — drei enthalten nur Großbuchstaben. Die mit PolyFonts eingegebenen Zeilen sind nicht mit den Cursortasten editierbar. Außerdem wird die Arbeitsgeschwindigkeit des Programmes bei der Verwendung mehrerer PolyFonts stark verringert.

Einen Nachteil haben alle Fonts: Die deutschen Sonderzeichen wurden nicht implementiert. Im Moment bleibt einem nichts anderes übrig, als zumindest die Umlaute »per Hand« mit Hilfe von zusätzlich geladenen Grafiken zu erzeu-

IT PFIFF



Eine Schrift, die über einem IFF-Hintergrund liegt, wirkt für sich

gen. Der Video-Anwender aus Deutschland muß zunächst noch etwas auf eine umgesetzte Version warten.

Obwohl sich mit der Vielfalt der vorhandenen Zeichensätze schon einiges anstellen läßt, wird der Video-Titler erst durch die möglichen Veränderungen und Verfremdungen der Zeichen richtig interessant.

Videos im Neonlicht

So lassen sich die Zeilen mit einem Neoneffekt in verschiedenen Stärken, verschiedenen 3D-Effekten, einem »Prägeffekt« und einem Sternfiltereffekt versehen (Bild 1). Insgesamt stehen 19 »Spezialeffekte« zur Verfügung, von denen ein großer Teil sehr realistisch wirkt. Zusätzlich können die Textzeilen noch mit einem Schatten versehen werden, dessen Richtung veränderbar ist. Auch die Richtung und die Länge des 3D-Rückteils lassen sich einstellen. Wird einmal ein Effekt gewünscht, den es noch nicht gibt, so steht der »Expert-Modus« zur Verfügung, in dem durch Kombination der Spezialeffekte erstaunliche Variationen entstehen. Leider ist das diesbezügliche Kapitel in der Anleitung entschieden zu kurz.

Um die »Spezialeffekte« zu erzeugen, arbeitet das Programm natürlich sehr intensiv mit verschiedenen Farben. Normalerweise stehen (im 320 x 200-Modus) 32 Farben zur Verfügung. Wird der »Extra-Halfbrite«-Modus aktiviert, so sind 64 Farben verfügbar, wobei die neuen 32 Farben halb so hell wie die bisherigen sind. Alle Farben werden im Video-Titler, ähnlich wie bei anderen Programmen, in einer Farbpalette eingestellt. Hier gibt es neben den üblichen Einstellmöglichkeiten eine Neuheit: Neben dem bewährten, aber umständlichen Einstellen der Farben mit den RGB-Schiebern (das HSV-System fehlt leider), läßt sich durch einen Mausklick eine Farbleiste anzeigen, die im HAM-Modus alle verfügbaren Farben darstellt.

Mit der Maus wird nun in den Farbbereich »gezoomt«, um genau die gewünschte Farbe zu erhalten. Zusätzlich enthält die Farbpalette noch eine »Spread«-Funktion, mit der sich sehr schöne Farbverläufe erzeugen lassen. Diese sind zum Beispiel für die Neon-Funktion unbedingt notwendig. Mit Hilfe der Farbpaletten werden die Farben von Buchstabe, 3D-Effekt und Schatten (auch für jede Textzeile getrennt) ausgewählt.

Insgesamt sind die Funktionen zur Bearbeitung und Manipulation von Texten wirklich sehr gut gelungen. Die Arbeitsgeschwindigkeit des Programmes ist aber nicht immer zufriedenstellend. Besonders ärgerlich ist es, daß beim Umschalten zwischen den einzelnen Textzeilen mit der Maus jedes Mal der gesamte Bildschirm gleich zweimal neu aufgebaut wird. Das liegt daran, daß zum Anklicken einer neuen Zeile die gerade bearbeitete erst einmal mit der rechten Maustaste deaktiviert werden muß. Der Bildschirm wird aber beim Deaktivieren und beim Anklicken einer Zeile jeweils neu aufgebaut. Dadurch kann sich die Arbeit bei der Verwendung mehrerer PolyFonts mit Spezialeffekten wie Neon sehr in die Länge ziehen. Da man zur Verwendung eines Menübefehls ebenfalls die rechte Maustaste benötigt, woraufhin der Bildschirm erst einmal wieder neu aufgebaut wird, wurden die meisten Menübefehle über die Tastatur zugänglich gemacht (Shortcuts). Das führt dazu, daß das Programm über siebzig Tastenkombinationen unterscheidet, die oft aus drei Tasten zusammengesetzt werden. Die beigelegte Tastaturreferenzkarte ist deshalb in den ersten Tagen das wichtigste Arbeitsmittel.

Damit der Hintergrund für den Video-Titel nicht so langweilig wirkt, können beliebige IFF-Bilder (außer im HAM-Modus erstellte) von Diskette in den Hintergrund geladen werden. Die Bilder sind mit Hilfe des »Clipboard«, einem Zwischenspeicher für Grafiken, leicht zu verzerren oder zu kippen, womit sich erstaunliche Effekte erzielen lassen. Zusätzlich kann das Clipboard automatisch mit einer Farbabstufung gefüllt werden.

Geschwindigkeit zu gering

Die »Paste«-Funktion dient dazu, die Grafik aus dem Clipboard in die Hintergrundgrafik »einzukleben«. Mit Hilfe eines der immerhin neun Menüpunkte der Paste-Funktion lassen sich beispielsweise eingegebene Schriftzüge mit einer Grafik unterlegen oder Grafiken auf die volle Bildschirmgröße bringen, was besonders im Overscan-Modus praktisch ist. Die Vielseitigkeit der Paste-Funktion muß allerdings mit einer relativ geringen Geschwindigkeit bezahlt werden. Ein weiteres Problem ist, daß bei eingeschalteter Paste-Funktion die eingegebenen Schriftzüge vorübergehend nicht mehr sichtbar sind. Bei manchen Bildern wird der Paste-Rahmen unsichtbar. Abhilfe wird durch eine Neueinstellung der letzten Farbe in der Palette geschaffen. Außer dem Clipboard-System gibt es zur Grafikbearbeitung noch einige Menüfunktionen, mit denen Grafiken auf ein Viertel verkleinert oder an die verschiedenen Auflösungen des Amiga angepaßt werden können.

Beim Laden von IFF-Bildern gibt es ein kleines Problem: Da das geladene Bild natürlich eine eigene Farbpalette hat, werden solche Effekte wie Neon, die eine Farbabstufung aus sechs Farben benötigen, nicht mehr richtig dargestellt. Hier muß man eben mit der Farbpalette herumprobieren, um zu

einem gelungenen Ergebnis zu kommen. Eine bessere Methode ist, mit dem Hilfsprogramm »Butcher« die Farbpalette des zu ladenden Bildes auf 16 Farben zu reduzieren.

Hat man schließlich eine Titelgrafik produziert, so wird sie als IFF-Bild auf Diskette gespeichert. Danach kann sie mit »Video Seg« weiterverarbeitet werden. Um das Ganze aber noch professioneller erscheinen zu lassen, kann man Bildersequenzen als ANIM-File (Animation) speichern und von Video Seg wieder in schneller Folge abspielen. Die im Video-Titler verwendete Technik der Animation ist allerdings außerordentlich umständlich: Jedes einzelne Bild muß mit der Hand erstellt und dann per Tastendruck auf Diskette gespeichert werden. Eine Korrekturmöglichkeit gibt es nicht. Da der Video-Titler beim Bildschirmaufbau und dem Verschieben von Textzeilen nicht gerade schnell ist, wird somit das Produzieren einer längeren Animation zum Geduldsspiel, besonders da man bei einem Fehler wieder von vorne anfangen darf.

Nachdem die Grafiken und Animationen, aus denen der Titel zusammengebaut werden soll, auf der Diskette stehen, wird nun das Video Seg-Programm geladen. Sein wichtigster Bestandteil ist ein sogenannter »Script-Editor«, in dem die Animationen und Bilder eingegeben werden, die der Computer anzeigen soll. Besonders interessant ist, daß das von Video Seg verwendete Animationsformat zu dem von VideoScape 3D kompatibel ist. Somit lassen sich auch VideoScape-Animationen problemlos einbauen. Man muß aber unbedingt darauf achten, daß alle verwendeten Animationen und Bilder dieselbe Bildschirmauflösung verwenden. Im schlimmsten Fall stürzt Video Seg ab, falls diese Bedingung nicht erfüllt ist.

Als besonderen Gag bietet Video Seg insgesamt 24 verschiedene Methoden an, wie ein Bild in ein anderes überblendet wird. Das Spektrum reicht vom einfachen Umschalten über ein weiches Aus- und Einblenden bis hin zum blockweisen Umschalten in einem

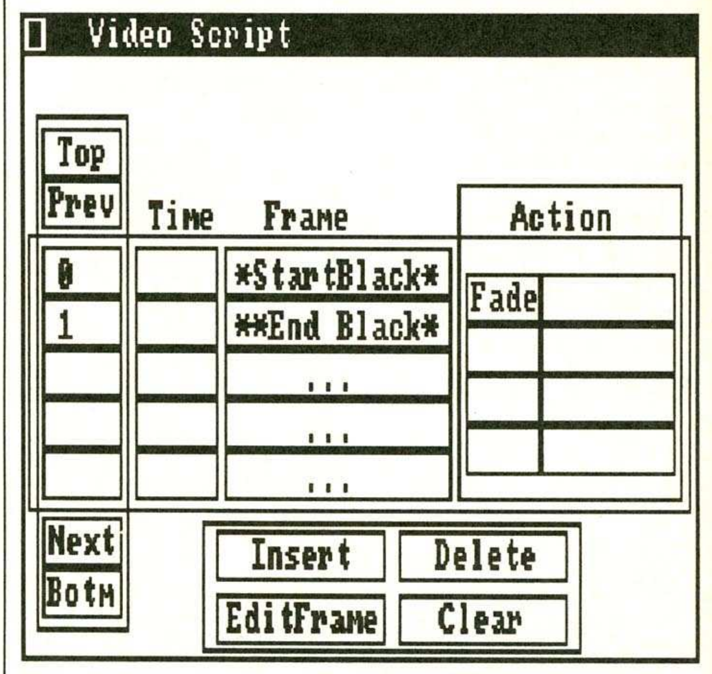


Bild 2. Der »Script Requester« des Video Seg-Programms

Nach einigem Experimentieren können sich die Ergebnisse aber durchaus sehen lassen. Die Animationen können entweder mit dem Video Seg (Bild 2) oder dem Public Domain-Programm »ShowAnim« zum Leben erweckt werden. Im Gegensatz zu den Informationen in der Anleitung lassen sich übrigens bereits mit 1 MByte Hauptspeicher Animationen erstellen.

Schachbrettmuster; Abwechslung ist garantiert. Für jedes Bild wird genau festgelegt, wie viele Sekunden es auf dem Bildschirm bleiben soll. Wenn die Ladezeiten von der Diskette zu lange dauern, können Bilder auch in einen der je nach Speichergröße bis zu 99 verfügbaren Puffer geladen und erst später auf den Bildschirm gebracht werden, wobei die Bilder und Animationen ge-



Bild 1. Eine Auswahl der verfügbaren Fonts und Effekte

packt im Speicher stehen. Das Script kann man natürlich auch auf Diskette speichern. Zwei Funktionen, die man sich noch wünschen könnte, sind Ein- und Ausblenden eines Titels, ohne dabei die Hintergrundgrafik abzuschalten und eine Möglichkeit, ähnlich wie beim Abspann im Kino, viele Zeilen von unten nach oben durchlaufen zu lassen.

Beide Programme, Video-Titler und Video Seg, sind grundsätzlich nicht allzu schwer zu bedienen, wenn auch die vielen Tastatur- und Menükommandos des Video-Titler erst nach einiger Übung zu beherrschen sind. Leider fehlen Sicherheitsabfragen, so beim Menübefehl »New« und bei den Tastenkommandos zum Löschen einzelner Textzeilen. Besonders in der Lernphase kann dies zu Ärgernissen führen. Die Anleitungen, besonders die des Video-Titlers, sind ausführlich und verständlich geschrieben.

Für einige etwas kompliziertere Funktionen gibt es sogar »Mini-Tutorials«, mit deren Hilfe man schnell zu einem Erfolgserlebnis kommt. Praktisch wäre allerdings noch ein Ausdruck der mitgelieferten Zeichensätze gewesen, damit man sich nicht durch die Fonts auf Diskette wühlen muß. Das Bildschirmfoto des »Symbol«-Fonts auf der Tastaturreferenzkarte ist kaum zu erkennen. Beide Anleitungen sind bisher nur in Englisch erhältlich, wie auch die gesamte Benutzerführung der beiden Programme englisch ist.

Insgesamt ist der Video-Titler sicher für jeden eine lohnende Investition, der professionell wirkende Titel auf seine Videobänder bringen will. Trotz der langsamen Arbeitsgeschwindigkeit und der vielen Tastenkombinationen, wird jeder begeistert sein, der einmal die verschiedenen Spezialeffekte in Aktion gesehen hat. Video Seg mit seinen vielen praktischen Funktionen tut dazu ein übriges.

(Andreas Lietz/jk)

AMIGA-WERTUNG

Software: Video-Titler

8,8 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■	■	■
Bedienung	■	■	■	■		
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■	■	■	

Fazit: Der Video-Titler erlaubt einen sehr flexiblen Umgang mit verschiedenen Zeichensätzen und Schriftarten und bietet somit die Grundlagen für gute Video-Titel. Bedienung und Arbeitsgeschwindigkeit könnten allerdings noch verbessert werden.

Positiv: Amiga- und Colorfonts verwendbar; viele Spezialeffekte; Poly-Fonts vielfältig veränderbar; vielseitige IFF-Grafikverarbeitung; große Menge von Überblendtechniken; arbeitet im PAL-Modus; kein Kopierschutz.

Negativ: sehr viele Tastenkombinationen; oft zu langsam; keine deutschen Sonderzeichen; nur fünf PolyFonts enthalten mit wenigen Sonderzeichen; umständliche Animationsmethode; Animationen nicht korrigierbar.

DATEN

Produkt: Video-Titler
Preis: 228 bis 248 Mark
Anbieter: Aegis Development
Anbieter: Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081; Compustore, Fritz-Reuter-Str. 6, 6000 Frankfurt, Tel. 069/567399

Workbench-Renovierung

AMIGA test

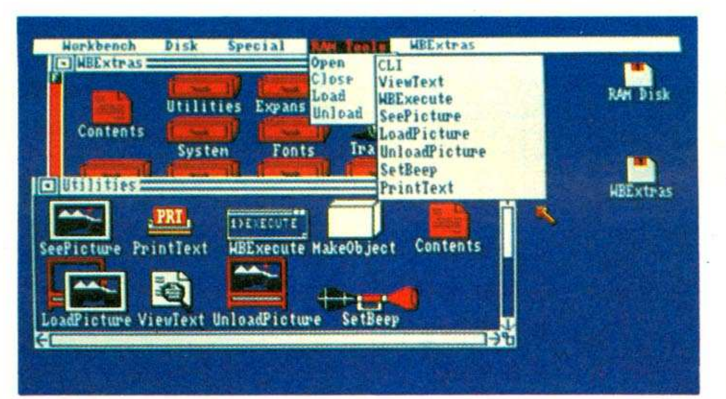
Die Amiga-Workbench ist zwar einfach zu benutzen, aber der ernsthafte Anwender kommt um das CLI nicht herum. »WBExtras« macht mit zwei neuen Menüs die Workbench vielseitiger und so die CLI-Benutzungszeit kürzer.

WBExtras steht für »Workbench Extras« und davon bietet das Programm einiges. Es zeigt, daß die Workbench auch für kompliziertere Aufgaben als bisher verwendet werden kann, wobei der Bedienungskomfort verbessert wird. Leider hat das Programm einen Paßwortschutz. Er hat eine Besonderheit: Sofern beim Start des Amiga das Datum korrekt eingestellt wurde, wird nur alle 90 Tage zur Eingabe eines neuen Paßwortes aufgefordert. Sobald das Programm läuft, bekommt die Workbench zwei neue Menüs: »WBExtras« und »RAM Tools«. Letzteres ist das Steuermenü für eine sehr interessante Funktion: das RAM-Tool-System.

Was aber ist ein RAM-Tool? Nichts anderes als ein beliebiges Amiga-Programm, das von der Workbench aus startbar ist. Jedes Programm dieser Art kopiert man mit Hilfe des »Load«-Befehles aus dem RAM-Tool-Menü in die RAM-Disk. Dazu muß man vorher das oder die zu kopierende(n) Programme anklicken. Ab sofort ist der Name aller kopierten Programme im Untermenü des »Open«-Befehles (Bild) zu finden, wo es jederzeit mit der Maus gestartet werden kann. Besonders interessant ist diese Technik natürlich für kurze Programme, die man während der Workbench-Arbeit immer wieder braucht, die aber in der »normalen« Workbench nicht vorhanden sind.

Von diesen Programmen werden immerhin acht mitgeliefert — unter anderem ein mausgesteuertes Tool zum Betrachten und Drucken von Textdateien sowie Programme zum Ansehen von IFF-Bildern. Die einzelnen Werkzeuge sind übrigens auch lauffähig, wenn WBE nicht aktiviert ist. Alle Tools werden entweder durch Doppelklick oder als RAM-Tool über das Menü aufgerufen. Bei den meisten von ihnen muß die zu bearbeitende Datei mit angeklickt werden. Was tut man aber, wenn das Programm kein Icon besitzt und nur vom CLI aus anzusprechen ist? Ganz einfach, man gibt dem Programm erst einmal ein Icon. Das erledigt das Hilfsprogramm »MakeObject«. Mit ihm läßt sich jede Datei auf einer Diskette mit einem beliebigen Icon versehen.

Ein besonders interessantes Hilfsprogramm ist »WBExecute«. Es startet CLI-Kommandodateien von der Workbench aus. Diese müssen lediglich (mit Hilfe von »MakeObject«) mit einem Icon versehen worden sein. Nun tippt man im »Info« des Icons noch »WBExecute« als Default-Tool ein. Dann läßt sich eine Kommandosequenz durch einen Doppelklick ausführen. Auf der Programmdiskette werden bereits zwanzig Sequenzen mitgeliefert. Leider stehen die von ihnen verwendeten Programme teilweise nicht auf der Diskette, so daß viele Sequenzen nicht verwendbar sind.



Die verfügbaren RAM-Tools im Untermenü

Ein kleines Problem haben die mitgelieferten Werkzeuge: Sie sind (noch) nicht auf die PAL-Version des Amiga zugeschnitten. Das führt dazu, daß sich zum Beispiel das Textfenster des »ViewText«-Tools nicht voll ausdehnen läßt und — wesentlich ärgerlicher — daß Grafiken in der Größe 320 x 256 und 640 x 256 im Interlace-Modus angezeigt werden, weil die Bildschirmhöhe als nicht ausreichend betrachtet wird. Die Programmautoren wollen allerdings noch in diesem Frühjahr eine deutsche Version ihres Programmes ausliefern, bei der dieser Fehler behoben werden soll.

Das RAM-Tool-System wäre ziemlich unsinnig, wenn man bei jedem Start von WBE alle Tools erst einmal »per Hand« von Diskette laden müßte. Deshalb wird in der Anleitung des Programmes beschrieben, wie die »Startup-Sequence« der Programmdiskette geändert werden muß, damit gleich beim Start die gewünschten Tools geladen werden. Beim Aufruf des WBE aus dem CLI werden die Namen der aufzurufenden Tools einfach hinter den Programmnamen geschrieben.

Mit dem RAM-Tool-System und den mitgelieferten Hilfsprogrammen arbeitet es sich insgesamt sehr gut. Besonders gut eignen sich die vielen auf Public Domain-Disketten verfügbaren kleinen Workbench-Hilfsprogramme. Durch eine trickreiche Programmierung schafft es WBE, beim Multitasking Speicherplatz zu sparen.

Das zweite Menü, »WBExtras«, enthält noch eine wichtige Funktion: Wird in diesem Menü »MultiSelect« angewählt, dient das als Ersatz für die SHIFT-Taste, die man normalerweise braucht, wenn man mehrere Icons gleichzeitig anklicken will. Der letzte Menüpunkt (»Reset«) schaltet WBE wieder ab. Leider klappt das aber nicht immer, denn falls noch irgendwelche Tools im Hintergrund laufen, wird die Abschaltung verzögert oder findet gar nicht statt. Das kann bis zum Absturz des Computers führen.

Die Anleitung des WBE ist ein achtzigseitiges Heft, in dem die Installation und die einzelnen Tools genau beschrieben werden. Der Anleitung fehlt es teilweise etwas an Übersichtlichkeit. Das Amiga-Glossar im Anhang ist dafür sehr gut gelungen. Zusätzlich gibt es auf der Programmdiskette noch eine Fülle von englischen Texten, in denen sich noch genauere Erläuterungen finden. Mehr für Programmentwickler enthält die Diskette einige Basic- und C-Programme, mit denen auf das WBE-System zugegriffen werden kann.

Insgesamt ist WBExtras sicher eine lohnende Anschaffung, wenn man der Workbench einige neue Möglichkeiten erschließen möchte. Die über das Menü aufrufbaren RAM-Tools sind bei der täglichen Arbeit wirklich praktisch. Ein CLI-Hilfsprogramm wie CLIMate oder Zing! kann WBExtras allerdings nicht ersetzen. (Andreas Lietz/jk)

AMIGA-WERTUNG

Software: WBExtras

7,8 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■		
Bedienung	■	■	■	■		
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■		

Fazit: Mit WBExtras wird die Arbeit mit der Workbench um einiges einfacher. Das RAM-Tool-System erlaubt einen sehr bequemen Programmaufruf; die mitgelieferten Hilfsprogramme enthalten Funktionen, die für jeden Amiga-Besitzer nützlich sind. Die CLI-Arbeitszeit wird wesentlich verringert.

Positiv: RAM-Tools schnell über Untermenü aufrufbar; CLI-Kommandodateien nun von der Workbench ausführbar; Tools von WBExtras unabhängig.

Negativ: Paßwortschutz; noch nicht an PAL angepaßt; mitgelieferte CLI-Kommandodateien zum Teil nicht lauffähig; bei Ausstieg nicht ganz absturzsicher.

DATEN

Produkt: WBExtras

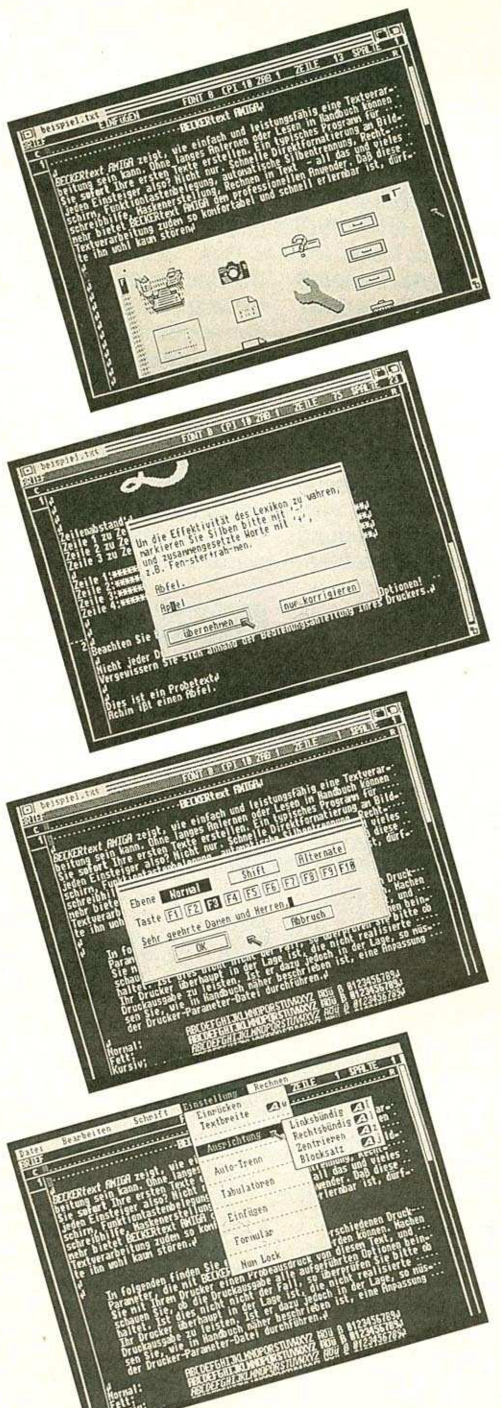
Preis: etwa 90 Mark

Hersteller: Lynn's Luna C

Anbieter: gutsortierter Fach- und Versandhandel.

WORT PERFEKT:

BECKERtext Amiga



Das deutsche Textwunder

Alle, die viel schreiben, brauchen eine Textverarbeitung, die alles kann und trotzdem schnell und komfortabel ist. Denn was nützt der größte Leistungsumfang, wenn man die Vielfalt der Funktionen nicht im Kopf hat und immer wieder das Handbuch wälzen muß?
Nein, eine Alleskönner-Textverarbeitung muß her. Mit allen Features, die man wirklich braucht, der vollen Integration in die AMIGA-INTUITION-Oberfläche – sprich: Anklicken aller Befehle mit der Maus – und dazu noch ein akzeptabler Preis. Wunschtraum oder Realität?
Die Antwort heißt BECKERtext AMIGA.

Schnelle Direktformatierung:
WYSIWYG-Prinzip: keine störenden Steuerzeichen im Text, schnelle Direktformatierung am Bildschirm mit allen Attributen (fett, kursiv, unterstrichen, Blocksatz, zentriert, linksbündig, rechtsbündig, hochstellen, tiefstellen, Horizontal- oder Vertikaldruck, Variation der Zeichendichte).

Einbindung von Grafiken:
Wenn schon AMIGA, dann auch eine Textverarbeitung, die Grafiken verarbeitet.
Für BECKERtext kein Problem: Das integrierte Hilfsprogramm BTSnap kann alle Grafiken im IFF-Format (Dateiformat, mit dem fast alle Mal- und Zeichenprogramme für den AMIGA arbeiten) und Bildschirmausschnitte der Workbench einlesen. Eine starke Sache.

Rechnen im Text:
Eine Textverarbeitung soll souverän mit Worten operieren, aber wie ist es mit Zahlen? Für BECKERtext AMIGA eine Leichtigkeit: Rechnen im Text, sowohl spalten- als auch zeilenweise. Mit bis zu 6 Nachkommastellen und 10stelliger Genauigkeit. Selbstverständlich mit Dezimaltabulator. Ein besonderer Vorteil für die Tabellenverarbeitung.

Formulare nach Wahl:
Mit BECKERtext AMIGA können Sie beliebige Formulare definieren und bis auf Abruf speichern (z. B. für Rechnungen, Lastschriftformulare, Tabellen, Briefpapier, Seitenlayout, etc.). Die lästige Neudefinition bewährter Standardformate entfällt – wieder ein Pluspunkt mehr.

Elektronische Rechtschreibhilfe:
Normalerweise folgt jeder Texteingabe die Korrektur. BECKERtext AMIGA leistet Vorarbeit: Das integrierte ONLINE-Lexikon überprüft den Text schon während der Eingabe auf Fehler in der Rechtschreibung (wahlweise auch danach). Da es individuell erweiterbar ist, eignet es sich auch für Fremdsprachen.

Überlegene Features:
Mehrspaltige Druckausgabe. Beim Ausdruck können Textdateien miteinander verknüpft werden. Multitasking: paralleles Arbeiten mit mehreren Programmen in verschiedenen Fenstern. Von einer Vorlage können bis zu 99 Kopien nacheinander ausgedruckt werden. Dreifache Funktionstastenbelegung mit maximal 160 Zeichen zur Speicherung von Floskeltexten oder Tastaturmakros. 1- und 2-bahniger Etikettenausdruck. Automatisches Erstellen von Stichwort- und Inhaltsverzeichnissen. Serienbrieffunktion mit Übernahmemöglichkeit aus beliebigen ASCII-Dateien. Datentransfer über RS 232. Umfangreiche Blockoperationen (Suchen, Ersetzen, Kopieren, Verschieben). Komfortable Druckeranpassung mit integriertem Treiber für alle gängigen Drucker. Querdruck auf Epson-kompatiblen Druckern bis zu 999 Zeichen pro Zeile. Ausführliches deutsches Handbuch. Minimalkonfiguration: 1 MByte RAM.

BECKERtext AMIGA
nur DM 199,–

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

COUPON

COUPON BITTE EINSENDEN AN:
DATA BECKER
MEROWINGERSTR. 30
4000 DÜSSELDORF

HIERMIT BESTELLE ICH

NAME, VORNAME

STRASSE, ORT

Aus den Tas

AMIGA test

Musikprogramme erfordern oft einiges an Notenkenntnis: Mit »Hotlicks« können auch Nichtmusiker ohne großen Aufwand musikalische Ideen verwirklichen.

Auf den ersten Blick erinnert das Programm stark an die auf dem Musikmarkt zahlreich vertretenen Personal-Keyboards. Doch bei genauerem Hinsehen stellt sich heraus, daß »Hotlicks« mehr als nur ein einfaches Simulationsprogramm ist. Die Bedienung erfolgt dabei natürlich mit der Maus. Mit ihr läßt sich sogar auf der groß dimensionierten Tastatur spielen. Über einem 6-Oktaven-Keyboard sind zwei Slider angeordnet, mit denen sich jeder seine ideale Tastaturbelegung einstellen kann. Mit den Slidern können zwei 1-Oktaven-Bereiche für die Amiga-Tastatur beliebig zugeordnet werden (Bild). Bei der Festlegung des Keyboards ist zu beachten, daß das Programm für den ersten 1-Oktaven-Bereich ein paar Spezialfunktionen anbietet.

4-Spur-Tapedeck

Durch Anklicken der »CHORD«-Funktion wird aus jeder gespielten Note ein dreistimmiger Dur-Akkord. Da der Amiga nur vier Soundkanäle zur Verfügung stellt, bleibt nur noch eine Stimme für die eigentliche Melodie übrig. In vielen Fällen ist das zu wenig — deshalb bietet »Hotlicks« die Möglichkeit Arpeggios einzusetzen. Bei diesem schon vom C 64 gut bekannten Trick wird ein Akkord in seine Einzeltöne aufgeteilt, die dann kurz nacheinander abgespielt werden. Der Vorteil dieser Methode liegt klar auf der Hand; es wird anstatt drei Stimmen nur noch eine einzige benötigt. Wie schnell die Arpeggios abgespielt werden, bestimmt der Tempo-Slider. Mit ihm werden auch die Aufnahme- und Abspielgeschwindigkeit des integrierten Musikrecorders festgelegt. Die Steuerung des Recorders erinnert an die Bedienung eines Cassettendecks. Ein Zählwerk hilft bestimmte Stellen eines Musikstücks besser wiederzufinden. Dies ist gerade bei der Korrektur von Aufnahmen enorm wichtig.

Hotlicks verfügt über 20 Soundspeicher. In jeden dieser Slots kann ein Klang, der dem IFF-8SVX-Format entspricht, geladen werden. Bei 512 KByte Hauptspeicher und umfangreicheren Sounds kann es allerdings vorkommen, daß nur zehn der 20 Speicher belegt werden können.

Das Einspielen eigener Stücke gestaltet sich relativ einfach. Als erstes kann eine der vier Spuren zur Aufnahme-Spur gewählt werden. Als nächstes wird das gewünschte Instrument in den Speicher geladen. Ein Klick auf den EDIT-Button läßt die Aufnahme beginnen. Als Hilfe tönt aus dem Lautsprecher ein Metronom, das durch das Aufblinken einer simulierten Leuchtdiode op-

AMIGA-WERTUNG

Software: Hotlicks

7,3 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■			
Dokumentation	■	■	■	■		
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■		

Fazit: Hotlicks ist ein Programm, das für den musikalischen Anfänger ohne Notenkenntnisse konzipiert wurde. Gerade für Anfänger wäre ein umfangreicheres Handbuch jedoch sehr wünschenswert gewesen. Behebt der Hersteller noch kleinere Mängel, so ist das Programm für alle Neu-Musiker eine Empfehlung wert.

Positiv: keine Notenkenntnisse erforderlich; Tastatur-Problem gut gelöst; kompatibel zu IFF-Standards; Verarbeitung von Script-Files.

Negativ: nicht absturzsicher; Handbuch zu knapp; mißlungene Rhythmen.

DATEN

Produkt: Hotlicks
Preis: 98 Mark
Hersteller: Infinity Software
Anbieter: Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081

ten gekitzelt



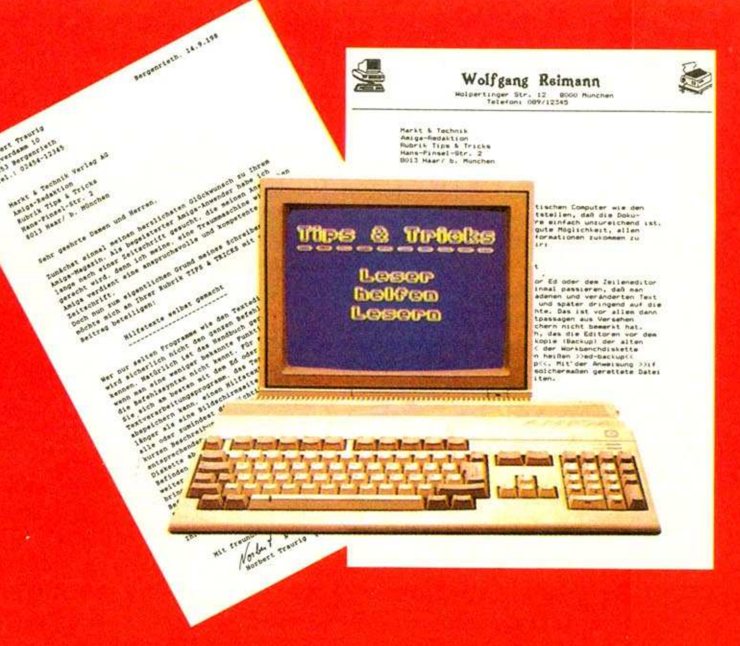
»Hotlicks« simuliert ein Keyboard auf der Amiga-Tastatur

tisch unterstützt wird. Das Einspielen der Noten erfolgt über die Amiga-Tastatur oder durch Anklicken der jeweiligen Noten auf dem Bildschirm-Keyboard. Dabei zählt das Zählwerk bei jeder Note je nach Dauer einen entsprechenden Wert mit. Die Pausen zwischen den einzelnen Noten werden allerdings nicht mitaufgenommen. Diese müssen später extra aufgezeichnet werden. Dazu dient die »1 Button Play«-Einrichtung. Mit ihr läßt sich eine bereits aufgenommene Spur Note für Note noch mal durchlaufen. Da mit diesem Programmteil aber nicht nur die Pausen, sondern auch alle Notenlängen neu eingegeben werden, hat man sehr sorgfältig zu verfahren. Meistens sind mehrere Anläufe nötig, um eine Spur mit korrektem Timing aufzunehmen. Bei den restlichen Spuren ist das Verfahren exakt das gleiche. Oft ist es bei der Aufnahme einer neuen Spur wünschenswert, gleichzeitig die bereits bespielten Kanäle zu hören. Das Anklicken des Accomp-Buttons stellt diese wichtige Funktion dem »Hot-Licks«-Anwender zur Verfügung. Jeder Benutzer kann sich seine eigenen Rhythmen erstellen und speichern. Dabei sind die zahlreichen Editierfunktionen ein großer Nutzen. Mit einer speziellen Funktion lassen sich Notenbereiche beliebig markieren. Da die Noten der einzelnen Tracks optisch nicht dargestellt werden, muß die Kennzeichnung eines Bereichs mit dem Gehör erfolgen. Die markierten Noten können mit den aus Textverarbeitungen bekannten Funktionen CUT, COPY und PASTE bearbeitet werden. Somit stellt das Löschen und Wiedereinfügen bestimmter Passagen keinerlei Probleme dar. Zahlreiche Musikstücke bestehen aus identischen Strophen oder einfachen Transponierungen. Auch diese Funktionen bereiten dem Anwender kein Kopfzerbrechen. Jeder Bereich kann um jeweils einen Halbton oder eine Oktave nach oben oder unten verschoben werden. Sollte eine Spur oder gar das ganze Stück nicht den ursprünglichen Vorstellungen entsprechen, kann mit »Clear Track« entsprechend gelöscht werden.

Ein spezielles Feature des Programms ist der Juke-Box-Mode. Bei Hotlicks lassen sich mehrere Songs durch ein Script-File gesteuert hintereinander abspielen. Dieses File kann mit jedem normalen ASCII-Texteditor erzeugt werden. Die wenigen Befehle der Script-Datei teilen dem Programm mit, in welchem Directory die Instrumente zu finden sind und welche Lieder gespielt werden sollen. Alles in allem wird das Bild von Hotlicks durch ein paar Kleinigkeiten etwas getrübt. Wem die herkömmlichen Musikprogramme eine Nummer zu groß sind, der sollte sich das Programm auf alle Fälle anschauen.

(B. Carli/jk)

Gesucht: Tips & Tricks zum Amiga



Halt, bevor Sie umblättern! — Haben Sie noch ein paar gute Tricks für den Amiga auf Lager? Dann sollte Sie unser Aufruf für die »Tips und Tricks« interessieren.

Schütteln Sie Ihre Asse aus dem Ärmel. Öffnen Sie Ihre Trickkiste für alle Leser. Senden Sie uns Ihre besten Ideen zum Amiga. Alle Griffe sind erlaubt.

- Hardware-Basteleien; genau richtig
- Software-Verbesserungen; fantastisch
- Anwendungs-Beispiele; super
- Spiele-Lösungen; oft der Retter in der Not
- Programmier-Kniffe; wunderbar
- Einsteigerhilfen; nicht wegzudenken
- Profi-Ratschläge; braucht jeder einmal

Es spielt keine Rolle, ob Sie selbst fortgeschrittener Programmierer oder ein Einsteiger sind. Sobald Sie etwas ausgeklügelt haben, schicken Sie es an uns. Wir geben Ihre Informationen weiter, damit Sie vielen Amiga-Fans helfen und neue Freunde gewinnen.

Für jeden Ihrer Beiträge, den wir veröffentlichen, erhalten Sie zusätzlich ein Honorar — damit Ihre Mühe auch belohnt wird. Also nichts wie ran an den Amiga. Experimentieren Sie, tüfteln Sie, suchen Sie nach den tollsten Tricks. Wenn Sie bereits ein paar gute Tips auf Lager haben — um so besser. Warten Sie nicht, bis ein anderer Leser auf dieselbe Idee kommt, denn jeden Beitrag können wir zunächst nur einmal drucken.

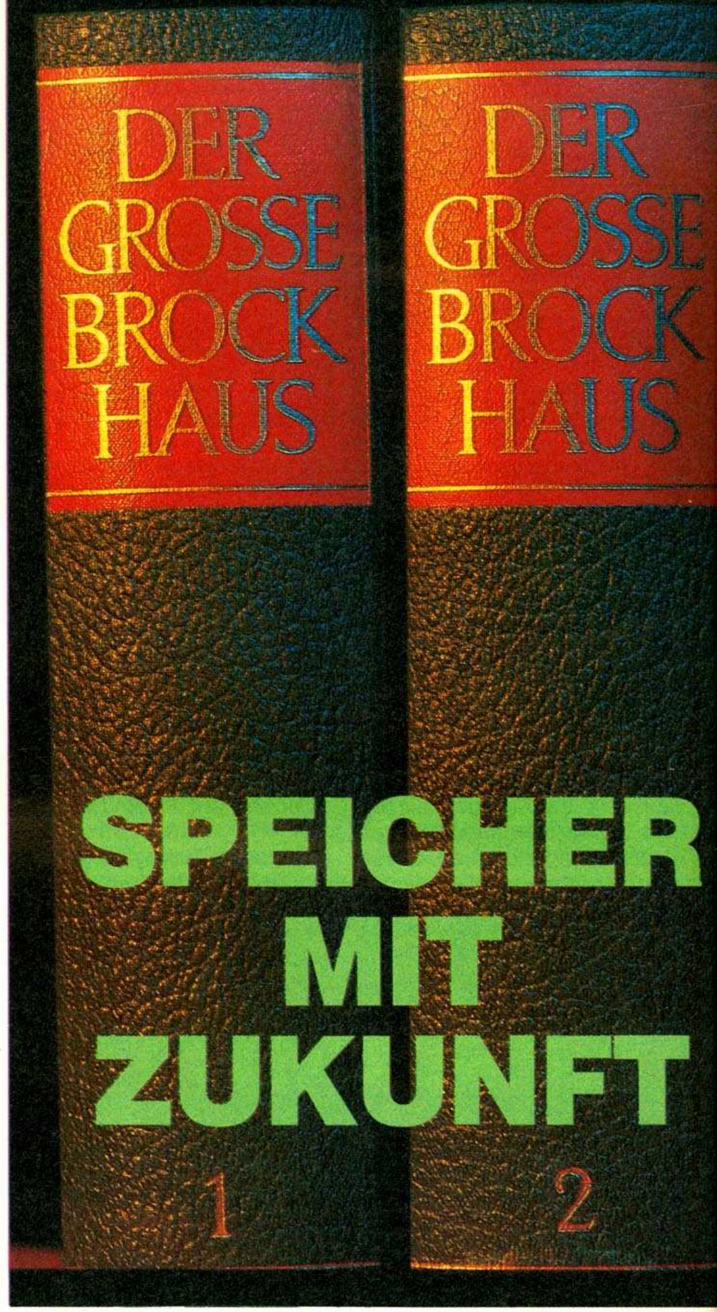
Schicken Sie Ihre »Tips und Tricks« an:

Markt & Technik Verlag AG
Redaktion AMIGA-Magazin
z.Hd. Ulrich Brieden
Aktion Tips & Tricks
Hans-Pinsel-Straße 2
8013 Haar bei München

Bei kurzen Texten reicht es, wenn Sie uns diese in einem einfachen Brief oder auf einer Postkarte zusenden. Bei Listings sollten Sie eine Programm-Diskette hinzufügen. Für Hardware-Basteleien sind Schaltpläne unbedingt erforderlich. Machen Sie also mit, erweitern Sie die Amiga-Fangemeinde durch Ihre Ideen.

(ub)

Festplatten sind externe Speicher mit hoher Kapazität. Wenn aber auch sie zu klein werden, muß ein Speicher her, der eine Menge an Daten faßt: das CD-ROM.



CD-ROMs sind in der Lage, gigantische Datenmengen in Größe

Spätestens seit dem Zeitpunkt, da die Hauptspeicher moderner Computer zu Größen von mehreren hundert KByte bis zu einigen MByte angewachsen sind, wird auch der Bedarf an externen Speichern, die diese Kapazitäten auch bewältigen können, immer größer. Einfache Diskettenlaufwerke reichen teilweise schon gar nicht mehr aus, diese Dimensionen aufzunehmen. Mit durchschnittlichen Kapazitäten von 300 bis fast 900 KByte sind die Diskettenlaufwerke oft überfordert. Es müssen also andere Laufwerke her, deren Fassungsvermögen hoch genug liegt. Die eine Alternative sind Festplattenlaufwerke, die zwischen 20 und 100 MByte Daten bei hohen Zugriffsgeschwindigkeiten fassen können. Jedoch sind diese Geräte noch immer relativ teuer (ab 1200 Mark). Für größere Kapazitäten bietet sich ein anderes Medium an: das CD-ROM (CD = Compact Disc). Quasi als Abfallprodukt der Unterhaltungsindustrie sind die leider nur lesbaren und nicht beschreibbaren CD-Platten für den Computer entstanden.

Schon seit einigen Jahren stehen in vielen Haushalten CD-Spieler mit den dazugehörigen, hochglänzenden Platten. Sie verdrängen immer mehr die lange gebräuchlichen Langspielplatten, da sie fast unanfällig gegen Kratzer oder Fingerabdrücke sind und außerdem einen viel besseren Klang bieten. Die Musik wird digital, also in Bits und Bytes verschlüsselt, auf der CD gespeichert und im CD-Player mit Digital-Analog-Wandlern in hörbare Klänge umgewandelt, womit wir bereits beim Thema sind: Die CD-Platte kann binäre Werte speichern, genau wie eine Diskette oder Festplatte.

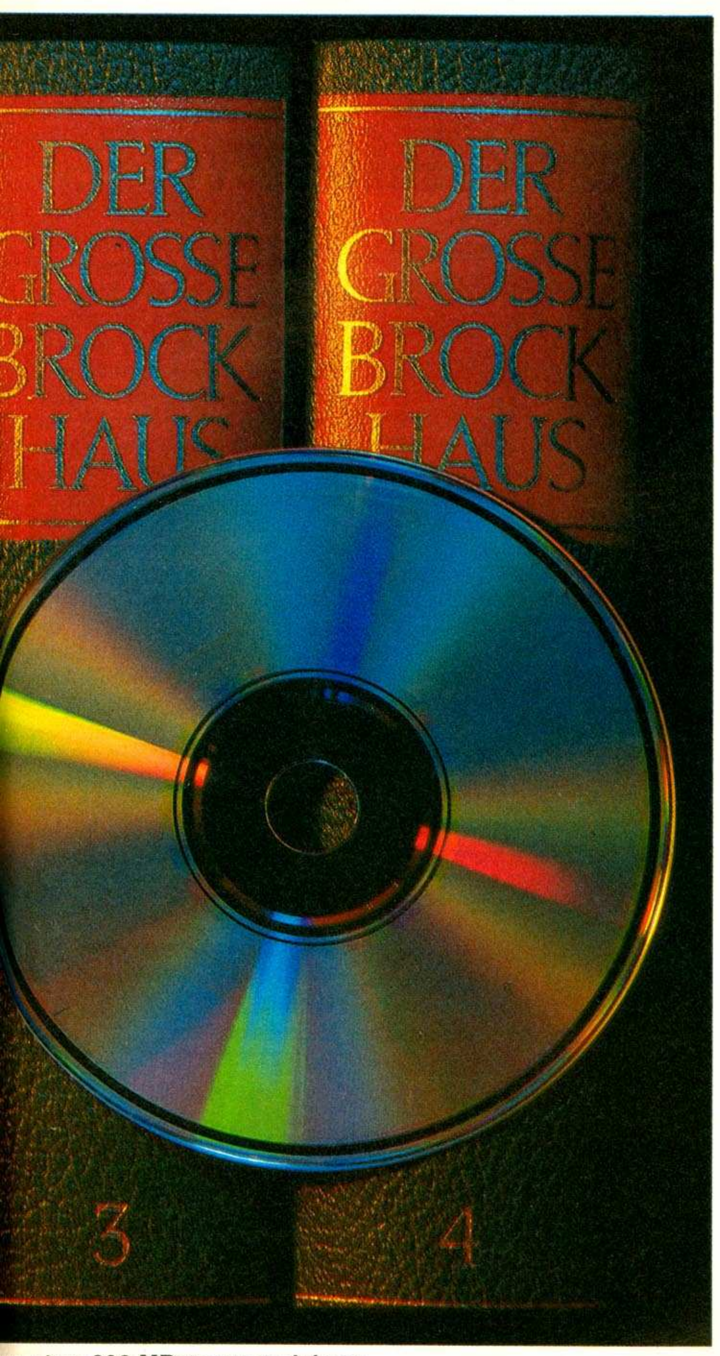
Compact Discs, die für den Computer lesbare Daten enthalten, erreichen heute schon Kapazitäten von über 500 MByte. Diese riesige Zahl an einzelnen Bytes können Sie sich vielleicht besser vorstellen, wenn Sie an über 200000 voll beschriebene Schreibmaschinenseiten oder an über 600 volle Amiga-Disketten denken.

Laser - Technik der Zukunft

Es ist schon eine ungeheure Menge an Informationen, die auf so einer kleinen Plastikscheibe Platz finden.

Aber wie findet diese Masse an Daten Platz auf dieser Scheibe und wie werden sie gelesen? Die Auflösung des Geheimnisses: Laser. Ein scharf gebündelter, winziger Laserstrahl tastet die Oberfläche ab. Dadurch können kleinere Spurbreiten als mit herkömmlichen Magnetköpfen für Disketten oder Festplatten erzielt werden. Die einzelnen Datenbits sind winzig klein auf der CD als Vertiefung oder glatte Fläche abgelegt, wobei eine Vertiefung eine binäre »0« und eine glatte Fläche eine »1« bedeutet. Beim Lesevorgang wird der Laserstrahl, je nach Beschaffenheit der Oberfläche, verschieden auf ein Fotoelement zurückgeworfen und so eine »0« oder »1« erkannt.

Wie bereits erwähnt, haben CDs eine viel höhere Datensicherheit als herkömmliche Magnetplatten und Disketten. Bei einer Magnetscheibe kann schon ein Fingerabdruck oder ein kleines Magnetfeld (etwa vom Lautsprecher oder Monitor) genügen, um einen Teil der Daten zu zerstören. Anders dagegen die Compact Disc. Der runde Plastik-Grundkörper, auf dem die Vertiefungen eingearbeitet sind, ist zusätzlich durch eine glatte, durchsichtige Schutzschicht gegen Beschädigungen oder Berührungen geschützt. Auch die bei Festplatten gefürchteten »Head-Crashs« (die Kollision des Lesekopfs mit der rotierenden Plattenoberfläche) kommen durch den Laserstrahl nicht mehr vor, ebensowenig wie mechanische Abnutzungen der Oberfläche durch den Lesekopf. Da auf einer CD die Daten durch mechanische Vertiefungen repräsentiert werden, machen der CD Magnetfelder nicht das Geringste aus. Ebenfalls können Fingerabdrücke der Datenschicht nichts anhaben. Auch Kratzer auf der Schutzschicht bedeuten noch lange nicht das Aus für die Daten. Zwar kann das Lesen der Daten beeinträchtigt oder gar behindert werden, aber bis zu einem gewissen Grad erkennt das hochempfindliche und präzise optische Abtast-Element (der Laser) trotzdem die richtigen Bits. Dies liegt am Laserstrahl, der durch optische Effekte quasi um den Kratzer auf der Schutzschicht herumgelenkt wird. Erst auf der verspiegelten Schicht trifft er schließlich mit einem mikro-



on etwa 600 MByte zu speichern

skopischen Lichtpunkt von 0,00003 mm Durchmesser auf die eigentlichen Daten.

Sollte der Kratzer oder die Beschädigung so schwerwiegend sein, daß die Oberfläche nicht mehr korrekt gelesen werden kann, ist noch nicht aller Tage Abend.

Strapazierfähige Scheiben

Durch aufwendige und durchdachte Checksummen sind trotzdem beeindruckend viele Daten rekonstruierbar. Für jeden der 2 KByte großen Datenblöcke existieren gleich mehrere Prüfsummen, die durch unterschiedliche Methoden ermittelt werden. Zur Datensicherheit sind die Bits eines Datenblocks nicht einfach sequentiell (der Reihe nach) angeordnet, sondern nach einem wohldurchdachten System auf mehrere Spuren verteilt. Dadurch ist die Wahrscheinlichkeit größer, daß durch die lesbaren Prüfsummen der ganze Datensatz errechenbar ist.

Einen Nachteil weist diese aufwendige Datenrekonstruktion allerdings auf: das Prüfen und Berechnen geht auf Kosten der Zugriffs- und Lesegeschwindigkeit. Die durchschnittliche Zugriffszeit eines CD-ROMs liegt bei etwa 150 ms und ist damit deutlich langsamer als bei einer Festplatte, die Zugriffsgeschwindigkeiten von 20 bis 80 ms erzielt.

Vorteilhaft beim CD-ROM ist auch, daß die Compact Disc genau wie beim Musikgerät auswechselbar ist. Die Herstellungskosten für eine CD-Platte liegen nicht hoch (etwa 5 bis 10 Mark). Teuer ist aber das Erfassen und das Aufbereiten der Daten. Zum Anlegen der umfangreichen Indextabellen und dem Erstellen der Prüfsummen ist ein Computer und ein großer Arbeitsaufwand nötig. Ebenfalls mit Computerunterstützung entsteht dann die Masterplatte, die als Vorlage für die Pressung dient.

CD-ROM-Laufwerke sind heutzutage vorwiegend in Verbindung mit Personal Computern im Einsatz.

CD-ROM-Datenbanken

Die einfach anzuschließenden Abtastgeräte werden von vielen Anbietern in unterschiedlichen Modellvarianten angeboten. Neben dem eigentlichen Abtastgerät ist zusätzlich eine Interface-Controllerkarte nötig. Neben Philips produzieren auch Firmen wie Hitachi, BCB Bertelsmann, Sony oder Control Data die zukunftsträchtigen Speichergeräte. Anwendungsgebiete sind etwa der Einsatz als Enzyklopädie, bei der der Anwender neben der gewohnten Textinformation auch auf Grafik-, Bild- und Toninformationen zurückgreifen kann. Eine weitere Anwendung ist die 1987 von Microsoft auf der »Infobase« (Informationsmesse) in Frankfurt vorgestellte integrierte CD-ROM-Anwendung »Bookshelf«. Auf der CD sind verschiedene Nachschlagewerke für den Schriftgebrauch am Schreibtisch, Grammatik, Satzbau, Umschreibungen, Stil und ähnliches gespeichert. Die zugehörige Software ermöglicht dem Anwender, die abgerufenen Informationen direkt von der CD in seine Textverarbeitung zu übernehmen, ohne das Programm wechseln zu müssen.

Es werden auch Versuche unternommen, auf CDs Stadtpläne zu verewigen, die dann mit Auto-Bordcomputern gelesen werden können. Sogar die Bibel ist schon erfaßt und liegt komplett auf einer einzigen Compact Disc vor. Oder die elektronische Enzyklopädie von Grolier, die auf einer einzigen Disc den gesamten Text des 20bändigen Werkes enthält — das sind 8 Millionen Wörter und 30 000 Einträge.

Wie anerkannt das Medium CD-ROM bereits ist, zeigt auch folgendes Beispiel: Seit bald 40 Jahren existiert der jährlich erscheinende Bezugsquellennachweis »Wer liefert was« des gleichnamigen Verlags; es ist das auflagenstärkste Industrie-Nachschlagewerk. In diesem, einem Telefonbuch ähnlichen »Buch« sind über 180 000 Produkthinweise in mehreren Sprachen enthalten, die auf zigtausende Firmen verweisen. Seit neuestem ist dieses Standardwerk für die Industrie auch auf CD-ROMs enthalten.

Die Möglichkeiten, die sich mit CD-ROMs anbieten, sind vielschichtig und werden wohl erst im Laufe der Zeit richtig entdeckt. CD-ROMs sind eines der Speichermedien der Zukunft. Für Personal Computer bereits erhältlich, können sie auch in naher Zukunft zu einem Preis von unter 1000 Mark für den Amiga erhältlich sein. Da die Kapazitäten Anwendungen wie große Datenbanksysteme oder ähnliche (Grafik-) Bibliotheken ermöglichen, ist wohl auch auf dem Amiga nicht das letzte Wort gesprochen. Wer weiß, vielleicht bringt der Markt in der nächsten Zukunft auch uns Amiga-Benutzern die ungeheuren Möglichkeiten der CD-ROMs näher, so daß der Amiga noch leistungsfähiger wird? (dm)

Die etwas andere Dateiverwaltung

AMIGA test

Im krassen Kontrast traditioneller Datenbanken steht der »Microfiche Filer«. Ein alternativer Weg zur Verwaltung Ihrer Daten?

Datenbanken im allgemeinen dienen der Aufbereitung und Katalogisierung großer Datenmengen. Die bisher gebräuchlichen Datenbanken sind in grafischer Hinsicht meist recht unattraktiv und von der Bedienung her ziemlich komplex. Manche dieser Programme erfordern das Erlernen eigener Datenbank-Programmiersprachen. Eine Datenbank für den Amiga könnte jedoch auch völlig anders aussehen. Wozu haben wir eine komfortable Benutzeroberfläche? Der Microfiche Filer von Software-Vision unterstützt die besonderen grafischen Fähigkeiten des Amiga. Die Grundidee des Programmes, ist den auf Zelluloid gebannten Microfiche-Datenblättern nachgeahmt. Sie kennen sicherlich diese kleinen Filmstreifen, die man unter einen Vergrößerungsprojektor legt. Ähnlich funktioniert der Microfiche Filer (MFF). Mit Hilfe der Maus bewegen Sie eine Art Lupe über ein kleines Datenfenster, das mit dem bloßen Auge nicht lesbar ist. Ein großes Fenster zeigt Ihnen den Inhalt der Daten im Lupenrahmen an. Gegenüber dem Filmstreifen hat der MFF den großen Vorteil, daß alle Daten frei veränderbar sind. Wie es sich für einen Grafikcomputer schickt, gehört zum MFF auch die Verwaltung von Bildern.

Die grafische Datenbank wird in einem zirka 120seitigen Ringbuch mit einliegender Programmdiskette geliefert. Die Anleitung ist reich bebildert und mit vielen Beispielen versehen. Wer noch nie mit einer Datenbank gearbeitet und noch wenig Erfahrung im Umgang mit dem Amiga hat, wird Dank der ausführlichen Anleitung gut mit dem Programm zurechtkommen. Einer Installation des Microfiche Filer auf einer Festplatte steht nichts im Wege, da das Programm nicht kopiergeschützt ist. MFF hatte seinen Ursprung als Public Domain-Programm, wodurch es nicht verwunderlich ist, daß ein ausführlicher Public Domain-Katalog als Musterdatenbank gleich mit auf Diskette geliefert wird.

Auf der Benutzeroberfläche des MFF muß man die Daten und Bilder mit der Lupe suchen

Ursprünglich Freeware

Gegenüber der Fish-Disk-Version sind jedoch viele nützliche Features hinzugekommen. Obwohl es sich um eine amerikanische Programmversion handelt, wird der deutsche PAL-Bildschirm automatisch in voller Größe ausgenutzt.

Eine Datenbank des Microfiche Filers setzt sich aus einer Ansammlung von Datensätzen zusammen, die aus mehreren Feldern bestehen können. Diese Felder enthalten beliebig Bilder, Zahlen oder Texte. Ein Textfeld kann eine maximale Größe von zirka 32 KByte besitzen. Leider unterstützt die Datenbank keine deutschen Umlaute. Gegen ein Umstellen des Zeichensatzes mit Hilfe des Setmap-Befehls wehrt sich das Programm hartnäckig. Bilder müssen im Standardformat ILBM gespeichert sein. Leider können Grafiken im HAM-Modus nicht verwaltet werden. Die Zahlen dürfen sich in einer Größenordnung zwischen plus und minus 21 Millionen bewegen. Bei der Verwaltung von Bildern ist es verständlich, daß die Grafiken nicht in voller Größe und unter Verwendung aller Bitplanes angezeigt werden können.

Datenreduzierte Bilder

Solange Sie mit dem MFF arbeiten, stehen Ihnen nur die vier Farben der Workbench zur Verfügung. Zum Verkleinern der Bilder werden diese entweder zusammengepreßt oder einfach abgeschnitten (siehe Bild). Für welche Methode Sie sich auch entscheiden, durch dieses Umwandlungsverfahren kann man das Datenbankbild manchmal nur schwer wiedererkennen. Erst beim Abrufen des Originalbildes wird der ursprüngliche Zustand wieder angezeigt.

Zur Eingabe der Daten gibt es einen Editor in dem das Datenfeld definiert wird. Hiermit geben Sie jedem Feld einen Namen und bestimmen mit dem Form-Editor den Umfang und die Abfragemodalitäten des Feldes. Die Daten einer Ebene werden untereinander und nebeneinander abgelegt. Sie können jedoch verschiedene Ebenen auszeichnen und so immer tiefer in einen Datensatz eindringen. Innerhalb der Datensätze ist als Rechenoperation nur die Summenbildung möglich, umfangreichere Rechenoperationen sind nicht implementiert. Die Druckroutinen des Microfiche Filers benutzen die Voreinstellungen der Workbench-Preferences. Sobald eine Grafik unter den Daten auftaucht, wird beim Ausdruck in den Grafikmodus umgeschaltet. Da der gesamte Datensatz ständig im Hauptspeicher gehalten wird, ist die Verarbeitung größerer Datenmengen nicht empfehlenswert. Andererseits sind die Suchroutinen im RAM besonders schnell. Ein Low Of Memory ist jedoch rasch provoziert. Hier stößt man auf die Grenzen des MFF. Es lassen sich keine Datensätze irgendwelcher anderer Datenbankprogramme laden, da der Microfiche Filer zu keinem dieser Programme kompatibel ist.

(Joschy Polierer/jk)

AMIGA-WERTUNG

Software: Microfiche Filer

6,2 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■				
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■			

Fazit: Durch die besondere Art der Verwaltung unterstützt dieses Datenbankprogramm die benutzerfreundliche Bedienung des Amiga in vollem Umfang. Kleinere Datensätze und Grafiken können bequem verwaltet und sortiert werden. Für eine kommerzielle Anwendung ist das Programm nur eingeschränkt zu empfehlen.

Positiv: PAL-Bildschirm; einfache Bedienung; kein Kopierschutz; integrierte Bilderverwaltung; schnelle Datensuche.

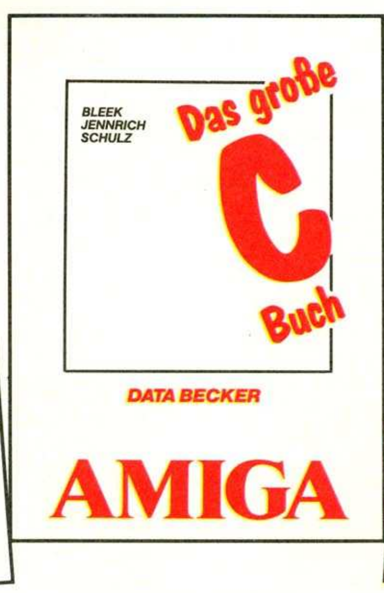
Negativ: keine Rechenoperationen; sehr speicherintensiv; kein Datenaustausch mit anderen Programmen.

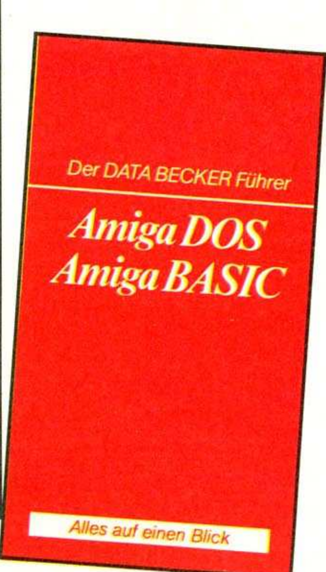
DATEN

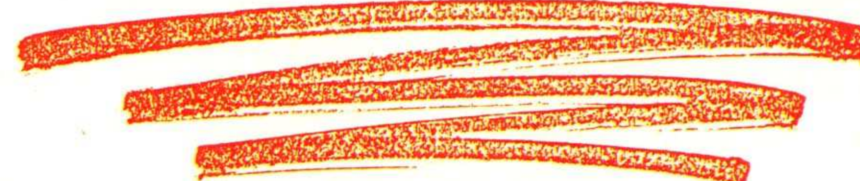
Produkt: Microfiche Filer
Preis: 190 Mark
Hersteller: Software Vision
Anbieter: Atlantis, Dunantstr. 53, 5030 Hürth, Tel. 02233/41081

Top aktuell:









Das Buch, das zur Amiga-Floppy keine Frage offenläßt. Hier finden Sie Dinge, die Sie in Ihrem Handbuch vergeblich suchen werden: Floppy-Operationen unter Workbench und unter AmigaDOS im CLI, relative und sequentielle Dateien, Aufbau der Diskette, Zugriff über Trackdisk-Device, Track lesen und schreiben, Kodier- und Dekodierroutinen des Betriebssystems – alles, was Sie zur Amiga-Floppy wissen müssen, wird hier detailliert und ausführlich beschrieben. Dazu eine Reihe nützlicher, hilfreicher Programme, die natürlich gleich auf Diskette mitgeliefert werden: z. B. ein Superkodierprogramm oder ein Floppyspeeder. Das Amiga Floppybuch – umfassendere Informationen werden Sie kaum finden.

Amiga Floppybuch
Hardcover, ca. 350 Seiten
inkl. Diskette, DM 59,-
erscheint ca. 3/88

Das große C-Buch zum Amiga – ein Buch für alle, die Spaß an C gefunden haben und nun darauf brennen, eigene, professionelle Programme zu entwickeln. Ausführlich erfahren Sie hier, wie ein C-Compiler arbeitet und wie Sie selbst die schwierigsten Probleme in C lösen – die beste Garantie, ein wirklich fehlerfreies Programm zu schreiben. Natürlich sollten Sie hierfür auch eine optimale Benutzeroberfläche entwickeln. Daher der zweite Schwerpunkt im großen C-Buch: die Programmierung von Screens, Windows, Pull-Down-Menüs und Gadgets. Gemeinsam mit den Autoren können Sie nun Ihr erstes großes Programmprojekt verwirklichen: einen C-Editor, der die Folding-Funktion („Wegfalten" von Textteilen) beherrscht. Wer mit diesem Buch arbeitet, wird in Zukunft höchstens noch eine Fehlerquelle haben: Tipfehler.

Das große C-Buch zum Amiga
Hardcover, ca. 600 Seiten
inkl. Diskette, DM 69,–
erscheint ca. 3/88

Der DATA BECKER Führer zu AmigaDOS und AmigaBASIC – ein nahezu unverzichtbares Nachschlagewerk für jeden Amiga-Anwender. Schnell und zuverlässig finden Sie hier jederzeit die Informationen, die Sie für Ihre tägliche Arbeit brauchen. Ob zu AmigaDOS oder zum AmigaBASIC – dieser DATA BECKER Führer läßt Sie nicht im Stich. Alle Befehle und Kommandos sind übersichtlich nach Sachgruppen, alphabetisch mit Kurzsyntax und nach Stichworten geordnet. Eben alles auf einem Blick. Ein Buch, das einfach zu jedem Amiga gehört.

Der DATA BECKER Führer
zu AmigaDOS &
AmigaBASIC
ca. 250 Seiten, DM 24,80
erscheint ca. 3/88

BESTELL-COUPON
Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1
Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten
unabhängig von der bestellten Stückzahl
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name
Straße
Ort

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Kombiniere, Desktop Video

AMIGA test

Die Einführung des Genlock von Futurevision stellt eine kleine Preissensation dar. Mit 500 Mark wird Desktop Video auch für den Heimanwender erschwinglich.

Videofilme mit Computerbildern zu überlagern, das ist eine ideale Verquickung zweier Hobbys. Seit es den Amiga gibt, stellen immer wieder einige Hersteller sogenannte Genlock-Interfaces auf dem Markt vor. Mit einem Genlock-Interface mischen Sie das Bild einer beliebige Videoquelle mit dem Amigabild. Hierbei handelt es sich natürlich nicht um eine Digitalisierung der Videobilder, sondern um eine echte Kombination der Videosignale. Dieses Verfahren kennen Sie vielleicht von den Tonmischpulten. Zwei Signale werden miteinander ins Mischpult eingespeist, vermischt und gemeinsam über einen Verstärker ausgegeben. Die Lautstärke der jeweiligen Signale kann unabhängig voneinander beeinflußt werden. So ähnlich funktioniert das mit dem Videosignal. Allerdings sind Videosignale viel komplexer aufgebaut. Sie lassen sich nicht so einfach kombinieren. Der Amiga hat jedoch den Vorteil, daß sich das Synchronisationssignal auch von außen zuführen läßt. Genau hierzu benötigen Sie ein Genlock. Die Videohardware des Amiga bekommt ihren Synchronisationsimpuls jetzt über das Genlock von einer extern angeschlossenen Videoquelle, zum Beispiel:

— einer Farbkamera,
— einem Camcorder oder
— einem Kabeltuner.

Wichtig ist nur, daß die Videoquelle ein gemischtes Signal mit der für die Synchronisation erforderlichen Information liefert. Da es sich hierbei um ein Standardsignal handelt, ist der Anschluß der Signalquellen in den meisten Fällen problemlos.

Das »PAL-Genlock« von Future Vision wird in einem kleinen etwa zigarrenschachtelgroßen Gehäuse geliefert. Auf der rechten Seite sind die Bedienungselemente herausgeführt. Der Anschluß erfolgt am RGB-Ausgang des Amiga. Der Stecker zum Monitor muß nun direkt am Genlock angeschlossen werden. Zwei Chinch-Buchsen dienen als Ein- und Ausgang für das externe Videosignal. Besondere Anschlußkabel liegen dem Gerät nicht bei. Dies ist auch durchaus sinnvoll, da der Hersteller nicht alle erdenklichen Normkupplungen beilegen



Bild 1. Videos und Texte: Ein Genlock macht's möglich

kann. Die Stromversorgung erfolgt direkt vom Amiga. Leider enthält die knappe Bedienungsanleitung keinerlei Hinweise über die Höhe der Stromaufnahme.

Film ab — läuft!

Direkt nach dem Einschalten arbeitet das Genlock ohne Schwierigkeiten. Auf dem Amigabildschirm erscheint das gemischte Videobild. Am Ausgang »Video-Out« steht Ihnen für einen zusätzlichen Aufnahmerecorder ebenfalls das kombinierte Signal zur Verfügung. Wir schlossen an diesen Ausgang einen zusätzlichen Monitor über den CVBS-Eingang zur Kontrolle an. Doch waren hier starke Qualitätsverluste gegenüber dem RGB-Monitor erkennbar. Dies liegt nicht nur an der überalterten PAL-Norm, sondern auch an der Ausgangsstufe des Genlocks.

Mit Hilfe des Fader-Reglers regulieren Sie stufenlos den Anteil zwischen Video- und Computerbild. Die Regler »Contrast«, »Illuminance« und »Colour« optimieren das Bild. So gelingt es mit Programmen wie »Aegis VideoTitler«, »TV-Text« oder »Deluxe-Paint« gute Ergebnisse zu erzielen (Bild 1). Bei dem Genlock handelt es sich um einen reinen Mischer. Sie können nur das gesamte Videosignal in das Computersignal einblenden. Effekte wie das Kopieren einzelner Farben oder die Negativdarstellung sind nicht möglich.

Der Aufbau des Genlocks macht einen guten Eindruck. Im Inneren sind alle Bauteile, einschließlich der Buchsen fest auf einer Platine montiert (Bild 2). Die Anschlußleitung zum Amiga ist jedoch deutlich zu kurz. Dies ist aufgrund der hohen Frequenz der übertragenen Signale erforderlich. Für den Anwender stellt sich jedoch das Problem, daß er bei jedem Nachregeln des Bildes hinter den Amiga »kriechen« muß. Die Drehregler bedürfen einer dringenden Verbesserung. Es handelt sich nur um einfache Plastikstangen, die noch dazu eng beieinander liegen. Wie schön wären hier richtige Schieberegler. Ansonsten ist das Genlock von Futurevision ein gelungener Wurf in die richtige Richtung.

(Joschy Polierer/ub)

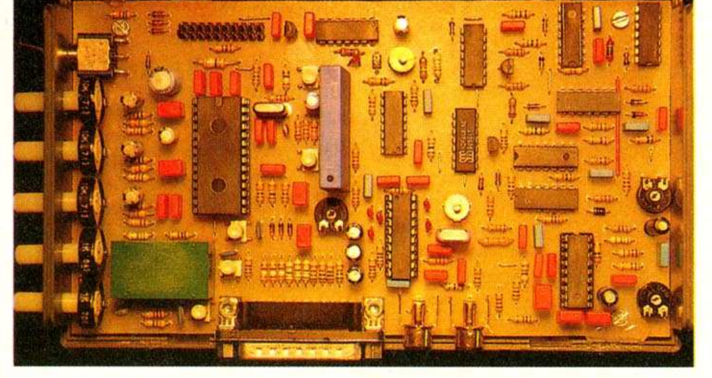


Bild 2. Das »Pal Genlock« ist sauber aufgebaut

Den für diese Ausgabe geplanten Testbericht eines Profi-Genlocks haben wir verschoben, da das von uns getestete IVS-1000 von Merkens mittlerweile in einer verbesserten Version geliefert wird.

AMIGA-WERTUNG

Hardware: FutureVision PAL-Genlock

7,6 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■			
Bedienung	■	■	■	■		
Verarbeitung	■	■	■	■		
Leistung	■	■	■	■		

Fazit: Das Genlock ist für den Heimanwender interessant, der die ersten Schritte in den Bereich Desktop Video unternehmen möchte. Er erhält ein Genlock, das zwar nur die reine Signalmischung erlaubt, aber zum Untertiteln eigener Videos geeignet ist.

Positiv: Preis; einfach zu bedienen; RGB-Bildqualität

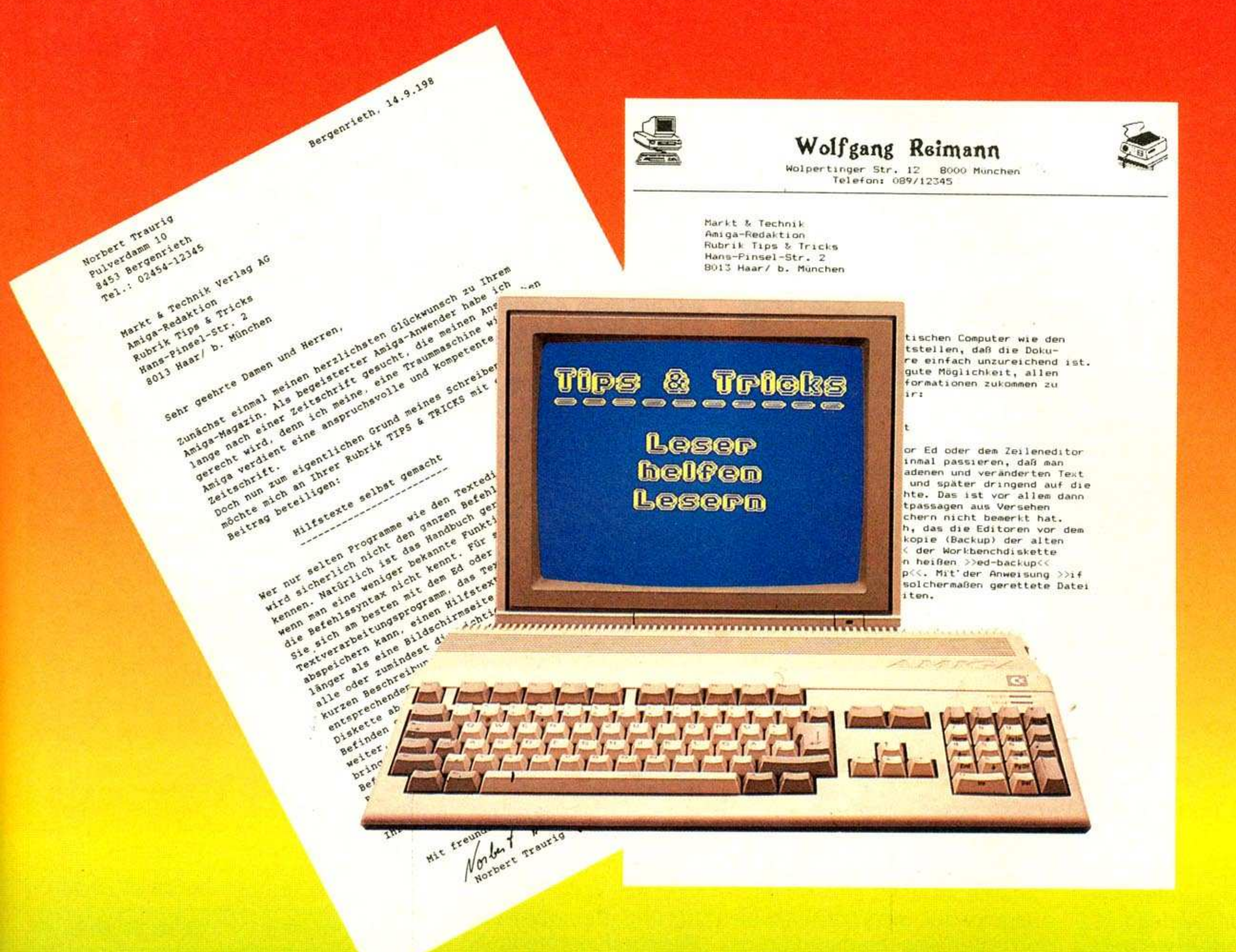
Negativ: kurze Anschlußleitung, Qualitätsverluste am CVBS-Ausgang, zu knappe Dokumentation.

DATEN

Produkt: PAL-Genlock
Preis: 498 Mark
Hersteller: Intelligent Video Systems
Anbieter: FutureVision, Michael Stutz, Friedrich-Veith-Str. 21, 6128 Höchst, Tel. (061 63) 1278

Gesucht:

Tips & Tricks zum Amiga



Sie sind ein begeisterter Amiga-Fan? Sie kennen Ihren Computer inzwischen so gut, daß die Begriffe »CLI«, »Amiga-DOS« oder »Workbench« für Sie keine Fremdwörter mehr darstellen? Dann beteiligen Sie sich doch an der Tips & Tricks-Rubrik »Ihres« neuen Magazins!

Die Tips & Tricks-Ecke im Amiga-Magazin ist eine reine »von-Lesern-für-Leser«-Rubrik. Das bedeutet, daß Sie durch Ihre Beiträge und durch Ihre guten Ideen anderen Lesern helfen können, mit dem neuen Computer besser zurechtzukommen. Sicherlich erinnern Sie sich noch an die Zeit, wo Sie begierig jede Zeile Information zum Amiga mit Heißhunger verschlangen. Gerade als Einsteiger ist man auf die Erfahrung und das Wissen von »Profis« angewiesen.

Sie haben eine neue Idee, wie man mit einem kleinen Amiga-DOS-Programm die Arbeit mit dem Computer vereinfachen kann? Einschicken!

Sie haben ein raffiniertes, kurzes Basic-, C- oder Assembler-Programm ausgeklügelt? Einschicken!

Sie haben einen Trick herausgefunden, wie man mit der einen oder anderen Software komfortabler arbeiten kann? Einschicken!

Sie haben Ihren Amiga durch eine kleine Bastelei aufgewertet? Einschicken! Jede gute Idee wird in »Ihrer« Rubrik veröffentlicht. Natürlich bekommen Sie dann von uns ein entsprechendes Honorar. Wichtig ist nur, daß Sie in einer kurzen Beschreibung den Grundgedanken Ihrer Idee zu Papier bringen. Wenn Sie ein Programm einschicken, sollten Sie nicht vergessen, eine Diskette mitzuschicken. Und sollte einmal ein Beitrag, aus welchen Gründen auch immer, nicht zur Veröffentlichung geeignet sein, bekommen Sie Ihre gesamten Unterlagen natürlich wieder zurückgesandt. Schreiben Sie an:

Markt & Technik Verlag AG
Redaktion Amiga-Magazin,
Aktion Tips & Tricks
Hans-Pinsel-Str. 2
8013 Haar bei München

Eines sollten Sie nicht vergessen: Sie müssen kein Vollprofi sein, um bei dieser Aktion mitzumachen. Auch Tricks, die für Sie schon längst zur Routine geworden sind, könnten anderen Lesern eine wichtige Hilfe sein. Vor allem kurze und leicht durchschaubare Listings sind begehrt. Wir wünschen Ihnen viel Spaß beim Programmieren und Knobeln! (ub)

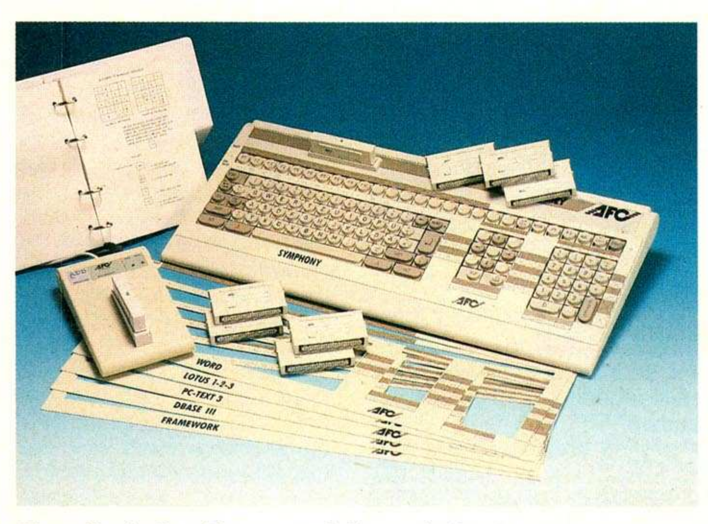
Der Verwandlungskünstler

AMIGA test

Um den Amiga sinnvoll auszunutzen, kann eine bessere, leistungsfähige Tastatur nur vorteilhaft sein. AFC bietet mit der AlphaKey-Tastatur eine leistungsfähige Alternative zur Standard-Tastatur an.

Die zum Amiga mitgelieferten Tastaturen sind zwar nicht die schlechtesten, jedoch können sie im Dauereinsatz nicht überzeugen. Die Amiga 2000-Tastatur erlaubt ja noch ein relativ angenehmes Arbeiten, aber die A1000-Tastatur schneidet da schon schlechter ab. Die Tasten liegen zu dicht aneinander und die ganze Tastatur ist zu klein. Wie schön wäre es da doch, mit einer vernünftigen Tastatur zu arbeiten...

Diesem Mangel setzt die neue AlphaKey-Tastatur von AFC Technology ein Ende, die sich sowohl an den A1000 wie auch an den A2000 anschließen läßt. Es müssen keine Kabel verändert oder spezielle Tastaturtreiber geladen werden. Neben den gewohnten Aufgaben einer Tastatur stellt sie aber noch weitere, sehr nützliche Funktionen zur Verfügung.



Die vollständige Tastatur mit Barcode-Leser

Freie Gestaltung

So lassen sich problemlos eigene Tastenbelegungen oder Makros (Befehlskombinationen) erstellen, die dauerhaft in einem EEPROM-Modul gespeichert werden können. Die Module sind austauschbar, so daß sich jeder Benutzer seine eigene Tastenbelegung erstellen und mitnehmen kann. Wahlweise kann ein Tastenklick (ein akustisches Signal) eingeschaltet werden. Zur Datenerfassung gestattet das Keyboard (Tastatur) den Anschluß externer Geräte (zum Beispiel Barcode-Leser), die auf seriellem Weg Zeichen an die Tastatur senden. Die ankommenden Signale werden von der Tastatur bearbeitet, gewandelt und als ASCII-Zeichen an den Computer gesendet, so daß vom Computer her keinerlei weitere Software zur Weiterverarbeitung nötig ist, was Kosten und Aufwand spart.

Die gesamte Tastatur ist mit Siemens-Tastenelementen ausgestattet, die sich durch einen guten Anschlag und Ergonomie auszeichnen. Das Gehäuse ist so geformt, daß der Benutzer eine gute Auflagefläche für die Handballen hat und so entspannter tippen kann.

Jede Sondertaste kann bis zu vierfach belegt werden. Auf jede Taste läßt sich ein beliebiges Zeichen oder eine Kombination aus Zeichen und Steuertasten (wie etwa <RETURN>) legen. In Kombination mit den Tasten <P1> und <P2> entstehen so drei Ebenen mit jeweils unabhängigen Belegungen. Die vierte Ebene, die mit der Taste <MODUL> erreicht wird, enthält die Belegung eines MS-DOS-Programms, das bereits auf dem Modul mitgeliefert wird. Zur Wahl stehen die Tastenbelegungen für Programme wie Word, Wordstar, dBase III Plus, Turbo-Pascal, Framework oder Symphony. Zu diesen Modulen liefert AFC auch entsprechende Tastaturschablonen, die nur auf das Keyboard aufgesteckt werden.

Flexibilität

Diese vierte Belegung ist sicher für alle MS-DOS-Benutzer auf dem Amiga (sei es mit dem Sidecar oder der PC/AT-Karte) eine spürbare Erleichterung. Praktisch ist auch, daß die Belegung der ersten drei Ebenen dauerhaft in das Modul gespeichert werden kann. Das Modul kann auch entnommen werden, so daß jeder Benutzer der Tastatur immer seine individuelle Belegung bei sich tragen kann.

Die Belegung der Tasten und das Speichern in ein Modul ist leicht zu erlernen und auch von jedem Nicht-Computerfachmann schnell zu begreifen.

Jedem Modul kann auch ein Paßwort zugeordnet werden, so daß jedes Modul nur von autorisierten Personen benutzt werden kann. Gibt der Anwender nicht beim Start das Paßwort ein, stellen sich die Tastatur und alle angeschlossenen Peripheriegeräte taub.

Eine weitere besondere Fähigkeit ist die Anschließbarkeit von Datenerfassungsgeräten. Über ein sogenanntes KCC-System (KCC = Key Code Conversation = Tastencode-Umwandlung) werden beispielsweise die Daten, die ein Strichcode-Leser erfaßt, in der Tastatur in Tastenimpulse verwandelt, das heißt die Tastatur simuliert eine Tasteneingabe. Eine Tastatureingabe wiederum wird von jedem Programm akzeptiert. Damit ist also beispielsweise jede Artikel- oder Lagerverwaltung, die mit Artikelnummern arbeitet, auf Strichcode umstellbar, ohne daß Software-Anpassungen notwendig sind.

Ein neu entwickelter Strichcode-Leser, der FDD Barcode-Reader, erlaubt zudem noch die Anpassung der vom Strichcode-Leser ausgegebenen Daten an die Bildschirmmaske des verwendeten Programms. So kann ohne Mehraufwand ein flexibles Datenerfassungssystem aufgebaut werden.

Das AlphaKey-Keyboard von AFC liegt preislich zwar in höheren Dimensionen (etwa 1600 Mark), jedoch kann diese Tastatur viel Arbeit abnehmen und auch bei der Datenerfassung Trümpfe ausspielen. Sei es mit dem ebenfalls nicht preiswerten Barcode-Leser (etwa 2500 Mark) oder durch andere Peripheriegeräte, die direkt an die Tastatur angeschlossen werden können. Dadurch, daß die Tastatur die komplette Steuerung übernimmt und keine Software angepaßt werden muß, amortisiert sich die Tastatur innerhalb kurzer Zeit. Auf jeden Fall ist sie aber eine gute Tastatur, die dem Amiga würdig ist. (dm)

AMIGA-WERTUNG

Hardware: AlphaKey-Tastatur

10,2 von 12

	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■	■	■

Fazit: Das Keyboard besticht durch ergonomische Gestaltung und leichten Tastenanschlag. Es läßt sich bequem damit arbeiten. Außerdem schafft die Möglichkeit, die Tastenbelegung frei zu gestalten, großen Komfort und Flexibilität

Positiv: ergonomische Tastatur und Tasten; guter Anschlag; frei belegbare Tastenbelegung (bis zu vierfach); Belegung auf Modul dauerhaft speicherbar; Peripherie anschließbar; auf Computerseite keine Zusatzsoftware nötig; für MS-DOS-Programme belegte Module erhältlich; deutsche Handbücher und Benutzerführung; Paßwortschutz; sowohl für A1000 wie auch A2000 verwendbar

Negativ: —

DATEN

Produkt: Alphakey-Tastatur

Preis: Tastatur 1390 Mark (+ MwSt.)
Barcode-Leser 2274 Mark (+ MwSt.)

Hersteller/Anbieter: AFC Technology, Bürgerbuschweg 48, 5090 Leverkusen 3, Tel. 02171/80057

Schnellzug

AMIGA test

Festplatten versprechen mehr Kapazität und schnellere Zugriffszeiten. Kann die X-Tension-Hard-Disk von Frank Electronic diesem Anspruch gerecht werden?

War es nicht schon immer Ihr Wunsch, Ihren Amiga durch externe Speicherkapazitäten auszubauen? Ein zweites Laufwerk erleichtert die Arbeit im Umgang mit den Dateien und Directories bereits sehr, eine Festplatte macht aus Ihrem Amiga ein professionelles Werkzeug.

Um Daten zu speichern, stehen Ihnen verschiedene Möglichkeiten zur Verfügung. Sie können zum Beispiel Ihre Informationen im Speicher des Computers ablegen. Wenn Sie die Stromversorgung unterbrechen, gehen alle Daten verloren. Diese Methode scheidet zur Langzeitsicherung also aus. Der Amiga verfügt aber glücklicherweise in der Grundausstattung über ein Diskettenlaufwerk, dessen Speicherkapazität einen beachtlichen Umfang von 880 KByte besitzt. Weitere extern angeschlossene Laufwerke erhöhen die Kapazität jeweils um den selben Betrag. Möchten Sie jedoch auf größere Datenmengen Zugriff haben, werden Sie sehr schnell an die Grenzen der Diskettenlaufwerke stoßen. Ein weiterer Nachteil ist, daß die Zugriffszeit auf die Daten der Disketten relativ langsam ist. Unter Zugriffszeit versteht man den Zeitraum, der verstreicht, bis die gewünschten Daten auf der Diskette gefunden und in den Computer geladen sind. Sie wissen sicher aus eigener Erfahrung wie lange es dauert, bis große Programme wie zum Beispiel Deluxe-Paint startbereit sind. Abhilfe verspricht hier eine Festplatte.

Speicherriese

Sie können sich die Festplatte (auch Hard-Disk genannt) wie eine riesige Diskette vorstellen, die ungeheuer schnell auf die entsprechenden Daten zugreifen kann. Tatsächlich besteht die Hard-Disk gleich aus mehreren großen »Disketten« und verfügt über mehrere Schreib- und Leseköpfe. Damit eine Festplatte die geforderte Geschwindigkeit erreichen und die gewünschten Datenmengen verwalten kann, benötigt sie einen eigenen Hardware-Controller. Beim Amiga 1000 zum Beispiel gibt es zwei Möglichkeiten, eine Hard-Disk anzuschließen. Entweder durch Einbau einer sogenannten Filecard (Festplatte mit Controller als Steckkarte) in das Sidecar, oder durch Anschluß einer speziellen Hard-Disk am Expansionbus. Der Weg, die Hard-Disk in Form einer Filecard über das Sidecar anzusteuern, hat den Nachteil, daß relativ lange Wartezeiten beim Booten (Neustart des Systems) in Kauf genommen werden müssen, da erst der PC-Teil gestartet werden muß. Schließt man jedoch die Festplatte direkt am Expansion-Port an, benötigt man zwar einen speziellen Controller, erreicht aber sehr viel höhere Geschwindigkeiten. Wir haben für Sie eine »echte« Amiga-Hard-Disk getestet, die »X-Tension HD1040« von Frank Electronic.

Diese Festplatte ist zum Anschluß sowohl am Amiga 1000 als auch, mit Hilfe eines Busumsetzers, der den um 180 Grad gedrehten Bus berücksichtigt, am Amiga 500 geeignet. Aus der X-Tension-Serie gibt es mehrere Festplatten mit Speicherkapazitäten zwischen 20 und 70 MByte. Die von uns getestete 40-MByte-Platte dürfte einen guten Kompromiß zwischen Wirtschaftlichkeit und Preiswürdigkeit darstellen. Die HD1040 macht einen guten äußeren Eindruck. Das formschöne Kunststoffgehäuse hat dieselbe Farbe wie der Amiga. Im Inneren des Geräts befindet sich außer der eigentlichen Festplatte die Controllerplatine und ein gekapseltes Schaltnetzteil. Durch die Verwendung eines relativ großen

Gehäuses konnte auf einen zusätzlichen Lüfter verzichtet werden. Die Verbindung mit der Außenwelt erfolgt über zwei Stecker. Der eine ist die Verbindung zum Stromnetz, der andere, »Host« genannt, führt über ein mehrpoliges Flachbandkabel an ein kleines Kästchen, das am Expansion-Port angeschlossen wird.

Leider ist dieser Port durch das Kästchen nicht mehr durchgeschleift. Es handelt sich damit um ein sogenanntes Endgerät. Ein Anschluß weiterer Endgeräte wie zum Beispiel des Sidecars ist nicht möglich.

Mit dem elektrischen Anschluß allein ist es noch nicht getan. Die Hard-Disk muß auch softwaremäßig in das Computersystem eingebunden werden. Da die Hard-Disk von Amiga-DOS nicht automatisch erkannt wird, ist im Lieferumfang der HD1040 ein Install-Programm enthalten, das auf die Workbench-Diskette kopiert werden muß.

Genauso wie eine Diskette muß eine Hard-Disk zuerst formatiert werden, bevor man Daten darauf ablegen kann. Dieser Formatierungsvorgang dauert natürlich bei 40 MByte Kapazität etwas länger als bei einer normalen 880-KByte-Diskette. Mit dem CLI-Befehl »MOUNT DH0:« wird die X-Tension nun in das System eingebunden. Dieser Befehl sollte am besten gleich mit in die Startup-Sequence übernommen werden, damit die Hard-Disk sofort nach dem Booten verfügbar ist. Jetzt kopiert man noch alles Nötige von der Workbench auf die Hard-Disk in entsprechende Verzeichnisse und erklärt die Directories mit dem ASSIGN-Befehl zu entsprechenden logischen Geräten. Ab jetzt greift Amiga-DOS direkt bei allen Befehlen auf die Festplatte zu, wodurch sich die Arbeitsgeschwindigkeit deutlich erhöht.

Geschwindigkeit ist überhaupt die Domäne der X-Tension-Festplatte. Mit einem atemberaubenden Tempo transferiert die HD1040 die gewünschten Daten. Vergleiche mit anderen Platten oder Filecards, die zum Beispiel im Sidecar eingebaut sein können, machen die Leistungsfähigkeit dieser Festplatte deutlich. Sie ist mit Sicherheit eine der schnellsten Hard-Disks, die für den Amiga erhältlich sind.

Benötigt man doch einmal wieder Daten von einem »normalen« Laufwerk, fällt der Unterschied in der Geschwindigkeit ganz besonders auf. Einen Wermutstropfen hat jedoch die Arbeit mit der Festplatte: Viele kopiergeschützte Programme sind nicht von der Festplatte lauffähig. Hier wird wieder einmal deutlich, wie ein Kopierschutz ein Programm für moderne Speichermedien unbrauchbar machen kann.

Die 40-MByte-X-Tension hinterläßt einen durchweg guten Eindruck, obwohl die mitgelieferte Dokumentation noch etwas ausführlicher sein könnte. Falls Sie zum Beispiel in einer Programmiersprache wie »C« programmieren, lassen sich durch die schnellen Datenzugriffszeiten auf diese Festplatte fast so schnelle Compilierungszeiten erreichen wie mit einer RAM-Disk.

Auf jeden Fall aber ist die Anschaffung einer Festplatte, ob nun mit 20 MByte, 40 MByte oder mehr Kapazität, sicher eine überlegenswerte Entscheidung, bedenkt man die größeren Kapazitäten und die damit verbundenen Arbeitserleichterungen. (Joschy Polierer/dm)

AMIGA-WERTUNG

Hardware:
X-Tension 40 MByte

8,9 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■		
Dokumentation	■	■	■	■		
Bedienung	■	■	■	■	■	
Verarbeitung	■	■	■	■	■	
Leistung	■	■	■	■	■	

Fazit: Die X-Tension-Festplatte gehört derzeit zu den schnellsten für den Amiga angebotenen Festplatten. Ihre große Kapazität und die große Zugriffsgeschwindigkeit machen sie zu einem Favoriten.
Positiv: gute mechanische und elektrische Verarbeitung; eigenes Netzteil; schnelle Zugriffszeit; sehr leise; Installationsprogramm im Lieferumfang enthalten.
Negativ: kein durchgeschleifter Bus; schlechte Dokumentation; großes Gehäuse

DATEN

Produkt: X-Tension 40-MByte-Festplatte
Preis: 20 MByte: 1498 Mark
40 MByte: 1998 Mark
72 MByte: 3498 Mark
Hersteller/Anbieter: Frank Electronic GmbH, Postfach 840073, 8500 Nürnberg 84, Tel. 0911/327732



AMIGA Markt&Technik

PROGRAMM-SERVICE

3-D-Landschaften aus dem Computer

Fraktalberge: Ein Muß für alle Fans von zufallserzeugten Grafiken. Fantastisch einfach in der Bedienung und sehr schnell. **Transfer:** Überträgt Bilder vom C64 auf den Amiga. Mit guter Software und leicht nachzubauender Hardware. **DiskSpy:** Direktes Ändern von Daten auf der Diskette ist mit diesem Werkzeug kein Problem mehr. Es stehen viele Befehle zur Verfügung. **ColorChange:** Ein Basic-Unterprogramm mit dem Sie einfach und schnell Ihre Wunschfarben auf beliebigen Bildschirmen einstellen können. **Troof:** Ein spannendes Spiel in Basic mit starker Grafik und vielen verschiedenen Levels. Außerdem finden Sie alle Programme auf Diskette, die im Inhaltsverzeichnis der Ausgabe 4/88 mit einem Diskettensymbol (■) gekennzeichnet sind.

Diskette für Amiga

Bestell-Nr. 48804 **DM 29,90 *** (sFr 24,90*/öS 299,–*)
* Unverbindliche Preisempfehlung

Bildschirmfüllende Boot-Bilder mit allen Extras

BootGirl: Fantastische Bilder sofort nach dem Reset. Bis zu 32 Farben mit Color-Cycling. Die Bilder können auch bildschirmfüllend ohne Rand sein. Ein absolutes Muß für jeden Amiga-Besitzer. **Cass-Cover:** Selbstgedruckte Kassettenhüllen geben Ihnen den richtigen Überblick. Einfache Bedienung macht das Eingeben und Ausdrucken zur wahren Freude. **Command:** Das Programm ermöglicht die Steuerung des Aztec-C-Compilers mit der Maus. Keine langen Eingaben per Tastatur, sondern ein einziger Mausklick startet nun die Übersetzung. **VideoText:** Ein unentbehrliches Werkzeug für alle Video-Fans, die ihren eigenen Vorspann mit dem Amiga generieren wollen. Laufbänder, verschiedene Schriften und IFF-Bilder sind nur einige Stichpunkte, die das Programm so interessant machen. Außerdem finden Sie alle Programme auf Diskette, die im Inhaltsverzeichnis der Ausgabe 3/88 mit einem Diskettensymbol (■) gekennzeichnet sind.

Diskette für Amiga

Bestell-Nr. 48803 **DM 29,90 *** (sFr 24,90*/öS 299,–*)
* Unverbindliche Preisempfehlung



Zeitschriften · Bücher
Software · Schulung

Weitere Angebote auf der Rückseite!

Markt&Technik Verlag AG, Buchverlag, Hans-Pinsel-Straße 2, 8013 Haar bei München, Telefon (089) 4613-0

Bestellungen im Ausland bitte an: SCHWEIZ: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 415656 · ÖSTERREICH: Markt&Technik Verlag Ges.m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 587 9455, Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526

ostscheckkonto Nr. des Absenders

mpfängerabschnitt

DM | Pf

r Postscheckkonto Nr.
4 199-803
ieferanschrift und Absender
er Zahlkarte

LZ Ort
erwendungszweck
M&T Buchverlag
rogramm-Service

Meine Kunden-Nr.:

DM | Pf | für Postscheckkonto Nr.
14 199-803

Absender der Zahlkarte

PSchA — Postscheckkonto Nr. des Absenders — Postscheckteilnehmer

Zahlkarte/Postüberweisung

Die **stark umrandeten Felder** sind nur auszufüllen, wenn ein **Postscheckkontoinhaber** das Formblatt als **Postüberweisung** verwendet (Erläuterung s. Rücks.)

DM | Pf | (DM-Betrag in Buchstaben wiederholen)

für Markt&Technik
Verlag Aktiengesellschaft
in 8013 Haar

Postscheckkonto Nr.
14 199-803
Postscheckamt
München

Ausstellungsdatum | Unterschrift

Für Vermerke des Absenders

Postscheckkonto Nr. des Absenders

Einlieferungsschein/Lastschriftzettel

DM | Pf

für Postscheckkonto Nr. — Postscheckamt
14 199-803 — München

für Markt&Technik
Verlag Aktiengesellschaft
Hans-Pinsel-Str. 2
in 8013 Haar

Postvermerk

PROGRAMM-SERVICE

Liga-Verwaltung für Sportprofis

LigaTab: Das Programm des Monats dient zur Verwaltung der verschiedensten Sportligen. Viele statistisch relevante Daten können komfortabel erfaßt und abgerufen werden. **Quatron:** Etwas ganz Besonderes für Spielefreaks ist dieses kurze C-Programm. Gute Grafik und hohe Spielmotivation sind seine Merkmale. **PlotIt:** Ausdrucke von Funktionen in fantastischer Qualität erhalten Sie mit diesem Basic-Programm. Die ideale Ergänzung zu dem Programm Kudiplo (Ausgabe 10/87). **MouseCreator und PointerMaker:** Ein Basic- und ein C-Programm zur Generierung von eigenen Mauszeigern. Damit kommt Leben in Ihre Maus. Natürlich sind auch alle weiteren Programme aus der Rubrik Programmieren der Ausgabe 2/88 auf der Diskette enthalten.
Diskette für Amiga

Bestell-Nr. 48802 **DM 29,90*** sFr 24,90*/öS 299,–*

Super-Spiel mit Sprache und toller Grafik

Domino: Ein Leckerbissen für alle Spiele-Fans ist dieses Programm des Monats. Sehr gute grafische Aufmachung und Sprachausgabe machen dieses Spiel besonders interessant. **Arraycomp:** Ein sehr nützliches Unterprogramm für alle Basic-Programmierer. Das Assemblerprogramm (in DATAs) findet gesuchte Strings bis zu 50mal schneller als ein Basic-Programm. **Julia:** Für alle Freunde schöner Grafik ist dieses Basic-Programm gedacht. Es stellt Juliamengen auf verschiedenste Weise auf dem Bildschirm dar. **Amiga-Sort:** Dieses Basic-Programm macht ein für allemal Schluß mit dem Chaos in Ihrer Diskettensammlung. Die Bedienung mit der Maus macht die Arbeit mit Amiga-Sort zum Vergnügen. **Disk.h:** Alle C-Programmierer brauchen dieses Programm, mit dem sogenannte »File Requester« durch einen Aufruf benutzt werden können. **Propgad:** Wer schon immer Schieberegler (Proportional Gadget) verwenden wollte, um Zahlen einzugeben, findet hier die nötigen Grundlagen.
Diskette für Amiga

Bestell-Nr. 48801 **DM 29,90*** sFr 24,90*/öS 299,–*

Super-Kopierprogramm mit viel Komfort

DCopy: Unser Programm des Monats, ein Kopierprogramm, das alles bietet, was man sich nur wünschen kann. Einige Fähigkeiten: Bis zu vier Laufwerke werden verwendet, Mehrfachkopien, abschaltbares Verify und vieles mehr. **SpeedHc:** Eine sehr schnelle Hardcopyroutine für Schwarzweißausdrucke mit höchster Qualität. Leicht an andere Drucker anzupassen. **Sternenhimmel:** Ein unentbehrliches Werkzeug für alle Himmelsbeobachter. Das Programm zeigt alle Sterne und Planeten von jedem beliebigen Punkt der nördlichen Hemisphäre. **Checkie42:** Der Checksummer für alle Programmiersprachen von Assembler über Basic bis zu C. Ab dieser Ausgabe finden Sie bei jedem Listing die Prüfziffern. **Joy:** Ein sehr kurzes und schnelles C-Programm zur Abfrage des Joysticks. Es ist leicht in eigene Programme einzubinden. **Amiga-Shell:** Ein C-Programm, das Komfort ins CLI bringt. Editieren der Befehlszeile, Funktionstastenbelegung und Aliasnamen sind nur einige Fähigkeiten dieses fantastischen Programms.
Diskette für Amiga

Bestell-Nr. 48705 **DM 29,90*** sFr 24,90*/öS 299,–*

IFF-Grafik-Show für Anspruchsvolle

IFF-Diashow: Als Resultat des in Ausgabe 6-7/87 gestarteten Malwettbewerbs präsentieren wir Ihnen zwei randvoll bespielte Disketten, die mit den besten Bildern des Wettbewerbs (siehe Amiga-Magazin 10/87, Seite 129) bespielt sind. Die Bilder können entweder mit dem enthaltenen Diashow-Programm angesehen oder mit jedem gängigen IFF-Malprogramm geladen werden. Lassen Sie sich überraschen, welche hochwertigen und vor allem sehenswerten Grafiken unsere begabten Leser geschaffen haben.
Zwei Disketten für Amiga

Bestell-Nr. 49705 **DM 29,90*** sFr 24,90*/öS 299,–*

Berechnung realistischer Bilder mit 3-D-Effekt

Raytracing: Mit dem Programm des Monats können Sie fantastische Bilder berechnen lassen, wobei der Strahlengang des Lichts berücksichtigt wird. **Termin:** Sie vergessen nie wieder Termine mit diesem Programm, das die Daten der nächsten 15 Tage automatisch anzeigt. **Chain Reaction:** Ein spannendes Spiel, das viel Taktik erfordert und praktisch nur auf dem Computer realisierbar ist. Lösen Sie eine Kettenreaktion aus! **EHB-Demo:** Wie man 64 Farben im LoRes-Modus benutzen kann, zeigt dieses C-Programm, das den Extra-Half-Brite-Modus verwendet. **Copper:** Auch von Basic aus kann der Copper für erstaunliche Effekte eingesetzt werden. Die Verwendung der nötigen Bibliotheken können Sie dabei erlernen. **Cursor Cycle:** Diese Utility macht Schluß mit dem unscheinbaren Cursor, indem es dessen Farbe ständig ändert. **MyCLI:** Eine unentbehrliche Hilfe für alle, die CLI-Fenster mit besonderer Voreinstellung benötigen. **MEd:** Das Schreiben von C-Programmen für Pull-down-Menüs verkürzt sich auf wenige Minuten. Mit einem weiteren Teil können Sie dann CLI-Befehle im Pull-down-Menü verwenden.
Diskette für Amiga

Bestell-Nr. 48704 **DM 29,90*** sFr 24,90*/öS 299,–*

Super-Malprogramme

Rainbow-Drawer: Dieses Programm des Monats bietet leistungsfähige Befehle und Funktionen, wie sie von professionellen Programmen bekannt sind: bis zu 32 Farben, alle Auflösungen, viele Befehle zum Zeichnen sowie FILL mit Mustern, BOW und anderem. **Turtle:** Mit dieser Befehlserweiterung verfügen Sie über die Grafikbefehle, die bei Logo bekannt und beliebt sind. **Fractals:** Dreidimensionale, realistische Gebirge mit Schattierung erzeugt dieses Programm. **Clouds:** Genauso wirklichkeitsnah wie die Gebirge, aber noch erstaunlicher, sind die Wolken, die Sie mit Clouds generieren. **Apfelmännchen:** Hiermit erzeugen Sie schöne Grafiken aus der beliebten Mandelbrot-Ebene. **Kudiplo:** Ein gutes, unverzichtbares Werkzeug für die Kurvendiskussion stellt »Kudiplo« dar. **Senso:** Testen Sie mit dieser Computer-Adaption des bekannten Spiels Ihr Gedächtnis! **Division:** Bis zu 32000 Nachkommastellen können durch dieses Programm berechnet werden. **Alert:** Alarme, zum Beispiel die bekannten Guru-Meditations, können Sie nun selbst programmieren. Das Programm ist in erster Linie für C-Programmierer aufschlußreich. **Border:** Lassen Sie den Fensterrahmen des CLI-Fensters einfach verschwinden! **SCD:** Mit diesem Utility können Sie den Pfadnamen in der Titelleiste des Fensters anzeigen.
Diskette für Amiga

Bestell-Nr. 48703 **DM 29,90*** sFr 24,90*/öS 299,–*

Disk-Operation vereinfacht

ExDir: Mit dem Programm des Monats der Ausgabe 8/9 sind alle Schwierigkeiten mit den Subdirectories endlich vorbei. **Hardcopy:** Diese nützliche Routine gestattet es, Teile des Bildschirms auszudrucken. **Bio:** Um Ihre körperliche und seelische Verfassung zu erfahren, können Sie dieses fantastische Biorhythmusprogramm einsetzen. **Mask:** Dieses Utility hilft Ihnen beim Herstellen komplexer Eingabemasken. **Freeman:** Den momentan verfügbaren Speicherplatz erfahren Sie mit dieser Routine im CLI-Window. **Vokabel:** Zum Lernen von Vokabeln ist dieses Programm ein sehr gutes Werkzeug mit viel Komfort. **SimpleSprite:** Wie man Sprites auf dem Amiga programmiert, sehen Sie in diesem Demoprogramm. **Vier gewinnt:** Wenn Sie gern spielen und strategisches Denken trainieren wollen, ist »Vier gewinnt« genau das Richtige für Sie. Weiterhin finden Sie auf der Diskette ein nützliches Tool zur Veränderung des CLI-Windows (InitCLI).
Diskette für Amiga

Bestell-Nr. 48702 **DM 29,90*** sFr 24,90*/öS 299,–*

* Unverbindliche Preisempfehlung

Sie suchen hilfreiche Utilities und professionelle Anwendungen für Ihren Computer? Sie wünschen sich gute Software zu vernünftigen Preisen? Hier finden Sie beides!
Unser stetig wachsendes Sortiment enthält interessante Listing-Software für alle gängigen Computertypen. Jede Woche erweitert sich unser aktuelles Angebot um eine weitere interessante Programmsammlung für jeweils einen Computertyp. Wenn Sie Fragen zu den Programmen in unserem Angebot haben, rufen Sie uns an: Telefon (089) 46 13-640 oder (089) 46 13-1 33.

Bei Fragen zu Bestellung und Versand der Programmservice-Disketten wählen Sie bitte Telefon (089) 46 13-232. Bestellungen bitte *nur gegen Vorauskasse* an: Markt&Technik Verlag AG, Unternehmensbereich Buchverlag, Hans-Pinsel-Straße 2, D-8013 Haar, Telefon (089) 46 13-0. Schweiz: Markt&Technik Vertriebs AG, Kollerstrasse 3, CH-6300 Zug, Telefon (042) 41 56 56. ÖSTERREICH: Markt&Technik Verlag Ges.m.b.H., Große Neugasse 28, A-1040 Wien, Telefon (0222) 587 94 55, Rudolf Lechner & Sohn, Heizwerkstraße 10, A-1232 Wien, Telefon (0222) 677526 Microcomput-ique, E. Schiller, Fasangasse 24, A-1030 Wien, Telefon (0222) 785661;. Bücherzentrum Meidling, Schönbrunner Straße 261, A-1120 Wien, Telefon (0222) 833196. Bestellungen aus anderen Ländern bitte nur schriftlich an: Markt&Technik Verlag AG, Abt. Buchvertrieb, Hans-Pinsel-Straße 2, D-8013 Haar, und gegen Bezahlung einer Rechnung im voraus.

Bitte verwenden Sie für Ihre Bestellung und Überweisung die abgedruckte Postgiro-Zahlkarte, oder senden Sie uns einen Verrechnungs-Scheck mit Ihrer Bestellung. Sie erleichtern uns die Auftragsabwicklung, und dafür berechnen wir Ihnen keine Versandkosten.

Einlieferungsschein/Lastschriftzettel
(nicht zu Mitteilungen an den Empfänger benutzen)

Gebühr für die Zahlkarte
(wird bei der Einlieferung bar erhoben)
bis 10 DM — — 90 Pf
über 10 DM (unbeschränkt) 1,50 DM
Bei Verwendung als Postüberweisung gebührenfrei

Bedienen Sie sich der Vorteile eines eigenen Postgirokontos
Auskunft hierüber erteilt jedes Postamt

Hinweis für Postgirokontoinhaber:
Dieses Formblatt können Sie auch als Postüberweisung benutzen, wenn Sie die stark umrandeten Felder zusätzlich ausfüllen. Die Wiederholung des Betrages in Buchstaben ist dann nicht erforderlich. Ihren Absender (mit Postleitzahl) brauchen Sie nur auf dem linken Abschnitt anzugeben.
1. Abkürzung für den Namen Ihres Postgiroamts (PGiroA) siehe unten
2. Im Feld »Postgiroteilnehmer« genügt Ihre Namensangabe
3. Die Unterschrift muß mit der beim Postgiroamt hinterlegten Unterschriftsprobe übereinstimmen
4. Bei Einsendung an das Postgiroamt bitte den Lastschriftzettel nach hinten umschlagen

Abkürzungen für die Ortsnamen der PGiroA:
Bln W = Berlin West, Kln = Köln, Dtmd = Dortmund, Lshfn = Ludwigshafen am Rhein, Esn = Essen, Ffm = Frankfurt am Main, Mchn = München, Nbg = Nürnberg, Hmb = Hamburg, Sbr = Saarbrücken, Han = Hannover, Stgt = Stuttgart, Krlh = Karlsruhe

Feld für postdienstliche Zwecke

Für Mitteilungen an den Empfänger

Bestellung Programm-Service — **Wichtig:** Lieferanschrift (Rückseite) nicht vergessen!

Bestell-Nr.	Anzahl	x Einzelpreis	= Gesamtpreis

Summe bitte auf Vorderseite übertragen — Gesamtsumme:

Frisch simuliert

Zu einer der interessantesten Anwendungen auf jedem Computer gehören Simulationen. Auch für den Amiga gibt es dabei nicht nur auf dem Spielesektor gute Programme. Wie wäre es beispielsweise mit der Simulation eines kompletten Planetariums. »Gallileo 2.0« ist das richtige für den angehenden Sternenforscher mit einer Datenbasis von über 1600 Sternen und dem »Yale Bright Star«-Katalog auf einer extra Diskette.

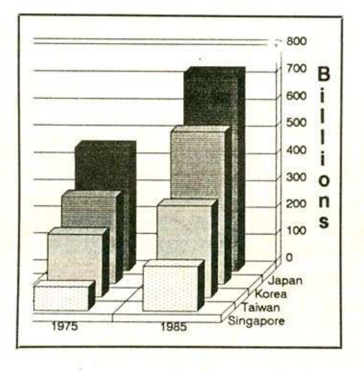
Hart an der Grenze zwischen Spiel und Wirtschaftssimulation bewegt sich »Ports of Call« von Aegis. Wenn Sie als Reeder Erfolg im internationalen Geschäft machen wollen, müssen Sie hier hart kalkulieren, immer die richtige Entscheidung treffen.



Schnelle Drucker

Wer beim Drucken auf Geschwindigkeit, Schriftqualität und Bedienungskomfort wert legt, kann auf den »Laserprinter 8« von Star oder den »DL3300« von Fujitsu zurückgreifen. Der Laserprinter 8 ist der erste Laserdrucker von Star. Er beherrscht die wichtigsten Drucker-Emulationen und besitzt bereits in der Grundkonfiguration 1 MByte an Speicher. Damit läßt sich schon einiges anfangen. Für Anwendungen im DTP-Bereich besitzt ein solcher Drucker wesentliche Vorteile gegenüber anderen Technologien.

Die Alternative ist ein 24-Nadel-Drucker: Der Fujitsu DL3300 ist ein Matrix-Drucker für gehobene Ansprüche. Er wartet mit einer angegebenen



Geschwindigkeit von maximal 288 Zeichen pro Sekunde auf. Ein kleiner Umbau macht den Drucker sogar farbtauglich. Doch das sind nur einige Details. Die Wahl, ob Sie sich für einen Laserdrucker oder einen 24-Nadel-Drucker entscheiden, der auch noch farbig druckt, fällt Ihnen leichter, wenn Sie den Test lesen.

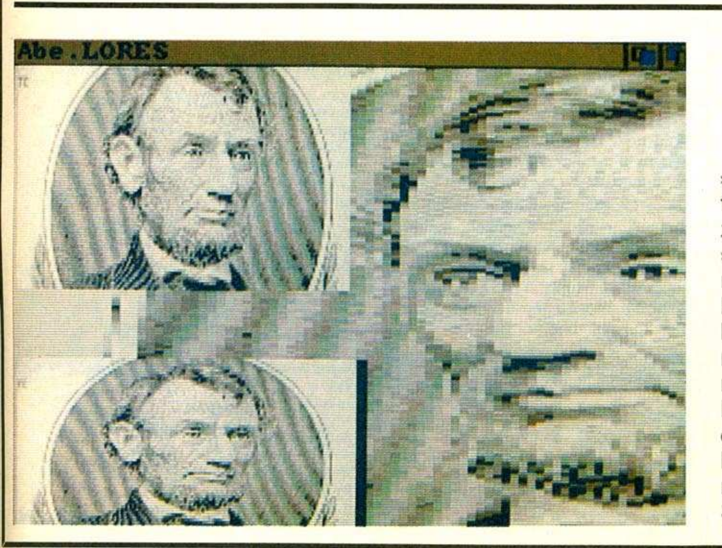
Hardware-Zusätze

Alle, die mit Hardware-Erweiterungen liebäugeln, finden in der nächsten Ausgabe Testberichte von zwei verschiedenen Amiga 500-Speichererweiterungen und dem A500-Subsystem. Interessant wird auch die »kleine« Turbo-Karte, die dem Amiga 2000 die zusätzliche 68020/68881-Leistung zukommen läßt. Für Bastelfreaks ist der Test eines neuen EPROMers gedacht, mit dem nun auch Amiga-Besitzer diese Bausteine programmieren können. Als Bauanleitungen präsentieren wir Ihnen ein Parallelport-Display und ein Speicheroszilloskop, mit dessen Hilfe der Amiga als Meßgerät eingesetzt werden kann.

Vorschau 5/88

AUSSERDEM IN DER NÄCHSTEN AUSGABE:
— DER GROSSE MESSE-BERICHT VON DER CEBIT: TRENDS, NEWS UND FAKTEN
— DISKETTEN-GRUNDLAGEN: DER STOFF, AUF DEM DIE DATEN SIND
— X-CAD: DAS SUPER KONSTRUKTIONS-PROGRAMM
— ENDLICH PROGRAMMIERBAR: DATENBANK AKQUISITION 1.3

Die nächste Ausgabe erscheint am 27.4.1988. Erhältlich bei Ihrem Zeitschriften-Händler.



Ein Kumpel für die Pixel

Bisher war das Grafik-Tool »Butcher« mit seinen Fähigkeiten zur Grafikmanipulation einzigartig auf dem Amiga. Mit »Pixmate« von Progressive Peripherals schickt sich ein Konkurrent an, die Vormachtstellung zu brechen. Durch eine neue Technik bei der Gräfikberechnung mit dem Blitter ist Pixmate sehr schnell geworden. Das Programm holt vor allem aus digitalisierten Bildern noch einiges heraus. Lesen Sie den ausführlichen Test.

Assembler-Fans ans Werk

Obwohl mit C für den Amiga eine leistungsfähige Programmiersprache zur Verfügung steht, geben die Assemblerfans nicht auf. Die Faszination, Speicher und Hardware direkt zu manipulieren, läßt sie nicht los. Damit Sie im Wettbewerb um das schnellste Programm nicht neue Dolchstoß-Legenden produzieren, führen wir mit einem neuen Kurs in die Feinheiten dieser Programmierung ein und helfen Ihnen, Assembler richtig anzuwenden.

Fortsetzung von Seite 35

Die ideale Datenbank

Projektes lädt das Programm eine Datei, die dazugehörige Maske und Indextabelle. Eine geöffnete Indexdatei wird vom Programm bei der Eingabe automatisch aktualisiert.

Anders als die anderen Kandidaten besitzt Organize! keine dynamische Speicherorganisation. Nicht voll ausgenutzte Felder werden mit Leerstellen aufgefüllt gespeichert. Zusammenfassung der Suchfunktionen: Bei aktiviertem Index findet »Browse« relativ schnell alle Datensätze, deren Indexfeld mit dem gesuchten Wert übereinstimmt. Die Menüfunktionen »Find next« und »Find previous« lokalisieren das nächste oder vorherige Auftreten. Die Funktionsvielfalt der Filterzeile steht nicht zur Verfügung. Mit Definition und Aktivierung eines von vier Filtern überprüft Organize! die Datei sequentiell auf Übereinstimmung mit den Filterkriterien. Auch hier lassen sich die beiden Find-Funktionen einsetzen.

Organize! wird weder mit deutschem Handbuch ausgeliefert noch reagiert es auf Umlaute oder kennt gar unser Zahlen- und Datenformat. Diese Mängel machen das Programm nur für diejenigen Anwender interessant, die für ihre Auswertungen Wert auf den großen Vorrat an mathematischen Funktionen legen.

In unserer Vergleichstabelle haben wir die wichtigsten Merkmale der besprochenen Datenmanager gegenübergestellt. Die Liste hilft Ihnen bei der Auswahl des richtigen Programms. Mit DB 2000 haben wir einen Standard vorgestellt, an dem sich kommende Datenbanken messen lassen müssen. Diese Aussage richtet sich nicht zuletzt auch an die Softwareentwickler. Die Programme müssen in Zukunft einfacher zu bedienen sein. DB 2000 ist dafür ein richtiges Konzept — der Amiga der richtige Computer.

Die Softwareentwicklung geht natürlich weiter. Superbase Professional, die programmierbare Version von Superbase, ist zur CeBIT angekündigt worden. Die Redaktion testet gerade die neueste Version von »Akquisition«, eine Datenbank aus den Staaten. Dieses Jahr verspricht interessant zu werden für die Datenbanker am Amiga. (pa)

AMIGA-WERTUNG

Software: Datenmanager Superbase

9,4 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	
Dokumentation	■	■	■	■		
Bedienung	■	■	■	■	■	
Erlernbarkeit	■	■	■	■	■	
Leistung	■	■	■	■	■	■

Fazit: Die zur Zeit leistungsfähigste Datenbank für den Amiga. Superbase kann durch die relationale Dateiorganisation bei Auswertungen mehrere Dateien verknüpfen. Für betriebliche/kaufmännische Anwendungen geeignet.

Positiv: Gruppenwechselverwaltung; durch relationale Dateistruktur flexible Auswertungsmöglichkeiten; einfache Bedienung;

Negativ: Reportausgabe umständlich; nicht programmierbar

DATEN

Produkt: Superbase
Preis: 248 Mark
Hersteller: Precision Software
Anbieter: Markt & Technik AG, Hans-Pinsel-Str. 2, 8013 Haar b. München, Tel. 089/46 13-0

AMIGA-WERTUNG

Software: Datenmanager GoAmiga! Datei

7,8 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■			
Dokumentation	■	■	■	■	■	
Bedienung	■	■	■	■	■	■
Erlernbarkeit	■	■	■	■	■	■
Leistung	■	■	■			

Fazit: Für den privaten Anwender mit beschränktem Datenbestand ein leistungsfähiges und einfach zu bedienendes Instrument zur Verwaltung von Karteien.

Positiv: Verwaltung von IFF-Dateien; einfache Bedienung; Datentransfer von und nach anderen Programmen; deutsches Handbuch.

Negativ: Datenbestand komplett im Speicher; keine komplexen Auswertungen möglich.

DATEN

Produkt: GoAmiga! Datei
Preis: 199 Mark
Hersteller: Softwareland
Anbieter: Softwareland, Franklinstr. 27, CH-8050 Zürich

AMIGA-WERTUNG

Software: Datenmanager Datamat

9,5 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■	■	■	■
Dokumentation	■	■	■	■	■	■
Bedienung	■	■	■	■		
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■	■	■	

Fazit: Die zur Zeit leistungsfähigste Dateiverwaltung für die Verwaltung umfangreicher Datenbestände. Durch Sonderfunktionen als einfache Textverarbeitung nutzbar. Für betrieblich/kaufmännische Anwendungen nur bedingt geeignet.

Positiv: sehr guter Maskeneditor; auf Einsteiger abgestimmte Dokumentation; einfache Bedienung bei der Eingabe; flexible Suchfunktionen; Hilfsdatei; Textverarbeitungsfunktionen; verwaltet Grafik/Sound; sehr preiswert

Negativ: Index nicht speicherbar; unvollkommene Gruppenwechselverwaltung

DATEN

Produkt: Datamat
Preis: 99 Mark
Hersteller: Data Becker GmbH
Anbieter: Data Becker GmbH, Merowinger Str. 30, 4000 Düsseldorf, Tel. 02 11/3 10 01 30

AMIGA-WERTUNG

Software: Datenmanager Organize!

5,8 von 12	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Preis/Leistung	■	■	■			
Dokumentation	■	■	■			
Bedienung	■	■	■	■		
Erlernbarkeit	■	■	■	■		
Leistung	■	■	■	■		

Fazit: Das Programm bietet nicht die Funktionsvielfalt anderer Programme. Sehr gut geeignet für Anwender, die neben der üblichen Dateiverwaltung auch Funktionen für die statistische Auswertung ihrer Daten benötigen.

Positiv: Datenkompatibilität zu dBase III; großer Vorrat mathematischer/statistischer Funktionen

Negativ: Dokumentation und Bedienerführung in Englisch; keine Gruppenwechselverwaltung; keine Verwaltung von IFF-Dateien; komplexe Auswertungen nicht möglich; Datentransfer zu anderen Programmen nur mit Hilfsprogrammen

DATEN

Produkt: Organize!
Preis: 198 Mark
Hersteller: Micro Systems
Anbieter: gutsortierter Fach- und Versandhandel

Inserenten

A + L Meier Vogt 103
AB-Computer Systeme 59, 60
AIT-User Group 62
Alcomp 135
Amigaland 125
Atlantis 51

Batavia 159
Bittendorf 59, 60, 62, 113
Blanke 95

CODAT 61, 107
Compu Store 75, 105, 107
Computing + Sound 75
CSJ Computersoft 93
C.S.S. 59
CSV Riegert 95
CWTG 95

Data Becker 31, 133, 141, 147
DTM 45, 121

Ecosoft 75
Edotronik 59
EDV Buchversand 95
Einstein Systems 62
Elektronik Zubehör 97, 117

Fastworks 61
Fischer, A. 61
Frank Elektronik 93
FSE Elektronik 65
Funkcenter Mitte 62
Future Vision 59

Gigatron 117
GIT Wiese 129
Grubert 119

Hagenau Computer 97
Hofstede 62
Huber, Angela 61

Intelligent Memory 55, 65, 110, 113
International Software Köln 75
Intersoft 113

Joysoft 60
Jumbo Soft 29
Juris, Klaus 61

Kingsoft 160
Kirschbaum 60
Knack Computer 119
Kröning 95
Kupke 2

Lamm 57
Ludewig + Wittwer 62

MAR Computer Shop 95
Markt&Technik Buchverlag 22, 26, 34, 138, 142, 152
Marvin 131
Mathes, Ernst 101
Medien-Center Rothholz & Möller 113
Message 85
Mükra Datentechnik 59
Musik- und Grafik-Software-Shop 62

NEC 17

Optivision 59
Ossowski 15

PD-Shop 61
PMD 59

Rainbow Data 61, 62
Rat + Tat 55
Reich Elektronik 61
Rossmöller 107
Computershop Ruth 60

SAFE B. Küppers & M. Aichinger 60
SAS Bernd 59
Scheer 97
Schmielewski 93
Schumacher, Chris 61
Sky Ware 60
Software 2000 65, 107
Soyka 33
Stalter, J.M. 89
Swiss Computer Arts 75
SWS Computer Systeme 60
Sybex Verlag 139
Syndrom Computer 39

Tröps Computertechnik 65

Vesalia Versand 71
Völkner 21
vts data 93

Waller, G. u. B. 65
WAW Elektronik 62
Weide Elektronik 55
Wolf, R. 60

Yellow Computing 55

Einem Teil dieser Ausgabe liegen Prospekte der Firma Neutronic, CH-Olten, bei.